付録 7

暗号技術調査 WG (軽量暗号) 報告書

暗号技術調査 (軽量暗号) ワーキンググループ

2015 年 3 月

目次

はじめに

本報告書は、暗号技術調査 WG(軽量暗号) が 2013 年度および 2014 年度に調査・検討した内容をまとめたものである。

1 章では、総括として、軽量暗号の現状と今後の活動方針をまとめている。

2 章では、軽量暗号に関する現状調査として、軽量暗号技術において、産業上のニーズがあり、具体的な暗号アルゴリズムの設計、安全性評価、実装評価が学会等で発表されている技術分類について代表的な軽量暗号アルゴリズムの現状調査（サーベイ）を行った結果をまとめている。

3 章では、軽量暗号に関わる新しい技術動向や関連する外部動向についての調査、軽量暗号の活用事例および標準化動向についてまとめている。

4 章では、軽量暗号のアプリケーションとして、自動車セキュリティおよび制御システムへの応用についてヒアリングを行った内容をまとめている。

5 章では、特に軽量ブロック暗号について、実装詳細評価を行った結果をまとめている。

以上の調査は、2014 年 12 月までに入手できる情報を対象とした。但し、2015 年 1 月に開催された 2015 年暗号と情報セキュリティシンポジウム (SCIS2015) で発表された内容を一部含む。

本報告書は下記に示す軽量暗号 WG 委員で執筆を行った。所属は 2015 年 3 月時点のものである。

主査	本間　尚文	国立大学法人東北大学　大学院情報科学研究科　情報基礎科学専攻　准教授
委員	青木　和麻呂	日本電信電話株式会社　NTT セキュアプラットフォーム研究所　主任研究員
委員	岩田　哲	国立大学法人名古屋大学　大学院工学研究科　計算理工学専攻　准教授
委員	小川　一人	NHK 放送技術研究所　ハイブリッド放送システム研究部　上級研究員
委員	崎山　一男	国立大学法人電気通信大学　大学院　情報理工学研究科　教授
委員	渋谷　香士	ソニー株式会社
委員	鈴木　大輔	三菱電機株式会社　情報技術総合研究所　情報セキュリティ技術部　開発第 1 グループ　主席研究員
委員	成吉　雄一郎	ルネサスエレクトロニクス株式会社　CPU システムソリューション部　主任技師
委員	峯松　一彦	日本電気株式会社　クラウドシステム研究所　主任研究員
委員	三宅　秀享	株式会社東芝　研究開発センター　コンピュータアーキテクチャ・セキュリティラボラトリー　研究主務
委員	渡辺　大	株式会社日立製作所　横浜研究所　エンタープライズシステム研究部　主任研究員

2

第 1 章

総括：軽量暗号の現状と今後の活動方針

1.1　CRYPTREC で扱う軽量暗号のスコープ

近年、リソースの限られたデバイスにも実装可能な「軽量暗号」(Lightweight Cryptography) の研究開発が進んでいる。これまで多くのアルゴリズムが発表され、国際標準化 (ISO/IEC 29192 など) も進んでいる。欧州では 2004 年から European Commission の第 6-7 次 Framework Programme の研究プロジェクト ECRYPT I, ECRYPT II のテーマとしても取り上げられてきた。日本も小型ハードウェア実装に適した暗号技術等で強みをもっている分野である。

低コスト・低消費電力で動作可能な軽量暗号技術は、今後もセンサー、車載機器、医療機器をはじめさまざまな用途での利用が期待されており、M2M (Machine to Machine), IoT (Internet of Things), CPS (Cyber Physical System) といった次世代のネットワークサービスを構築する上で有効なセキュリティ技術の一つとなることが期待される。

CRYPTREC では、主として電子政府で利用する暗号技術について検討を行っているが、電子政府のみに閉じることなく、さまざまな領域で利用される暗号技術についても技術調査を行い、社会に役立つ形で情報提供を行うことを目指している。軽量暗号技術が求められる製品やサービスにおいて、利用者が最適な暗号方式を選択でき、容易に調達できることを目指し、2013 年度より CRYPTREC 暗号技術評価委員会の下に軽量暗号 WG が設置された。

軽量暗号としてこれまで提案されてきた暗号技術には、ハードウェア実装のサイズ、消費電力量、組み込みソフトウェア実装で必要なメモリサイズ等さまざまな性能指標で最適化されたものがあり、「軽量暗号」に対して一般的に合意されている定義はない。また、性能と安全性のトレードオフもあり、実際には色々な扱いが可能な幅がある。本 WG では、以上の状況を鑑み、「実装性能と安全性のトレードオフを勘案した上で、従来の暗号技術に対して特定の性能指標で優位性（軽量性）を持つように設計された暗号技術」をスコープとし、用途が想定される代表的な性能指標に対して優位性を主張する暗号を主な対象とする。

また、現時点で、公開鍵暗号系において「軽量暗号」として広くコンセンサスがとれている方式はほとんどないため、本報告書では共通鍵暗号系の軽量暗号を対象としている。

軽量暗号が求められるアプリケーションでの要求条件のうち、本報告書では代表的な性能指標として下記に注目する。

- ハードウェア実装における
  - 回路規模
  - 消費電力量
  - レイテンシ（リアルタイム性能）
- 組み込みソフトウェア実装における
  - メモリサイズ（ROM/RAM）

ハードウェア実装の回路規模は、半導体のコストに直結し、また、消費電力 (Power) の指標にもなり得ることが知られている。回路規模の小型化は、RFID をはじめとする回路実装面積の要求条件が厳しいアプリケーションで重要な要件である。また、バッテリや外部供給電源がなく、電磁誘導等で駆動するデバイスにおいても重要な要件である。

消費電力量（Energy）の低減は、人体へ埋め込まれたり密着装備される医療機器をはじめ、バッテリで駆動するあらゆるデバイスで求められる要件である。

レイテンシ（遅延時間）は 1 回の暗号化（復号）処理に必要な時間である。低遅延性はメモリ暗号化や車載機器などのリアルタイム性が求められるアプリケーションで必須の要件である。

組み込みソフトウェア実装では、組み込みマイコン上で実現されるさまざまなアプリケーションの一部として、暗号機能を実装することが多い。組み込みマイコンでは、ROM や RAM のサイズが限られており、小さく実装できる暗号ほど、選択できるマイコンの品種が増えたり、コストを下げられる等の利点がある。組み込みマイコンは家電機器やセンサー、車載向け等で広く利用されており、実装に必要なメモリサイズ (ROM/RAM) が少ないことはこれらのアプリケーションで重要な要件である。

性能指標	アプリケーションの例
回路規模 (消費電力, コスト)	RFID、低コストセンサー
消費電力量	医療機器、バッテリ駆動デバイス
レイテンシ（リアルタイム性能）	メモリ暗号化、車載機器、産業向け I/O デバイス制御
メモリサイズ (ROM/RAM)	家電機器、センサー、車載機器

1.2 既存暗号に対して優位性をもつ分野

ここで性能指標の視点から軽量暗号が既存暗号に対して優位性を持ちうる分野について述べる。

LSI への実装を想定した回路規模の視点では、現在提案されている軽量暗号と AES の差は数 kgate 程度である。2014 年現在、モバイル向けの SoC (System on a Chip) や GPU で主流となっている 40nm 以下のプロセスで設計される LSI においては、典型的なダイサイズは 50mm^2 から 150mm^2 であり、1mm^2 あたり 1.6Mgate 程度も搭載できるため [1]、数 kgate 程度の回路規模が暗号の優劣の指標とはなりえない。これは、回路全体の 0.1% 未満に関するゲート数削減の議論となるためである。一方で、文献 [2] のミューチップのような、ダイサイズが 50μm 角 (250μm^2) クラスのチップでは数 kgate の差がクリティカルな課題となり、暗号機能の搭載可否に影響を与えうる。現時点では、このサイズのチップはアセンブリの難易度が高いため、500μm 角程度のチップが RFID では主流となっているが、このケースにおいても利用するプロセスが 180nm など古いプロセスであれば、数 kgate の回路規模の差が実装可否に影響を与える可能性がある。

また一般に、回路規模が小さいほど、消費電力あるいは消費電力量は減る傾向にある。環境発電に代表される低消費電力が求められるアプリケーションにおいては、様々な観点で低消費電力化を図る設計が必要となる。軽量暗号を利用することで消費電力あるいは消費電力量に関する設計条件を緩和する効果が期待できる。

次にレイテンシ（リアルタイム性能）の視点では、AES に対して 2 倍の応答速度をおよそ 1/10 の回路規模で実現できる軽量暗号が存在する [3]。この文献の例では、20kgate 程度の回路を用いれば 10ns 以下で暗号演算が可能とされている。一方、AES で同様のリアルタイム性能を得るためには、200kgate 使ったとしても 15ns 必要である。現時点で、産業向け I/O デバイス制御に代表されるような μs オーダーのリアルタイム性能が求められる通信路において暗号技術は利用されていないが、このようなアルゴリズムを利用することで、チップへのコストインパクトなしに暗号技術を利用できる可能性がある。

4

最後にソフトウェア実装における軽量暗号の性能指標について述べる。プログラムサイズの観点での AES に対する軽量暗号の優位性として、AES に対しておよそ 1/4 の ROM サイズで実装可能な軽量暗号が存在する [4]。この文献の例では、ルネサスエレクトロニクス社製の組み込みマイコン RL78 を用いた性能評価が行われている。RL78 は産業分野や自動車など幅広く利用されているマイコンの一つであるが、文献 [4] ではそのプラットフォーム上で 220 Bytes の ROM サイズで暗号演算が可能であることが示されている。長期間にわたって利用されてきたレガシー製品に対して、暗号機能を新たに搭載するといったアップデートを施す場合、残された ROM 領域に暗号を実装する必要があり、軽量暗号でなければ搭載できないケースが起こりうる。また、新規に暗号機能を搭載する製品を開発する場合であっても、暗号が使用する ROM 領域の削減が実現できれば、製品単価の安いチップを選定することができる。たとえば RL78 では、ROM サイズを 1KB から 512KB までの間から選ぶことができる。

2020 年にはセンサー 1 兆個、IoT 機器 500 億台の時代が到来すると言われており、前述のようなローエンドのマイコンが利用されている機器においても暗号技術が必要になることが予想される。また、自動運転が実用化され、工場やプラントがクラウドとシームレスにつながる時代が来ると予想されている。このような時代においては、現時点で暗号技術が利用されていない領域であっても、今後活用の必要性が高まると考えられる。軽量暗号は、現時点で暗号技術を搭載していない、あるいは実装上の制約から想定すらしていない機器やシステムにおいて、将来的に実装面での制約を緩和する効果を期待できる。

1.3 軽量暗号で達成可能な安全性

世の中に提案されている様々な暗号技術は様々な性能指標により評価できる。CRYPTREC では様々な暗号技術を評価し、CRYPTREC 暗号リストを維持している。CRYPTREC 暗号リストのうち、電子政府推奨暗号リスト及び推奨候補暗号リストは CRYPTREC により安全性及び実装性能が確認された方式である。これはカテゴリ毎で想定されている範囲でどのような利用がなされたとしても、安全性の問題が生じないとされており、速度などの実装性能についても実装環境毎の差が少ないバランスのよいものを意味している。もちろん、リスト中に注釈がついているものはその注釈の限定の範囲での話である。以下、この節ではそのような方式は議論対象外とする。

一方、軽量暗号は前節で述べられたように従来の暗号技術に対して特定の性能指標で優位性を持つように設計されている。それぞれの性能指標の間には一般にトレードオフが存在することから、提案されている軽量暗号の中には安全性が電子政府推奨暗号や推奨候補暗号より低くなっている方式も存在する。例えば関連鍵攻撃について安全かどうかは保証せず、その分、速度を稼いでいると主張している方式もある。とはいえ、利用場面によってはこのような高い安全性は不要であり、電子政府推奨暗号や推奨候補暗号では高い安全性が消費電力など別の性能指標の足を引っ張っている場合もあることから、安全性の一部に目をつぶった軽量暗号の方が有利な場合もある。よって軽量暗号は利用法によっては有効な技術であるが、設計者が主張するもしくは第三者による安全性評価結果については十分に注意する必要がある。

軽量暗号と謳っている方式の多くはハードウェア実装の回路規模が小さいものが多い。ブロック暗号を実装するためには、ブロック長のビット数に応じた中間状態を保持することが必須であるため、軽量ブロック暗号はブロック長として 128 ビットより小さなものが選ばれることが多い。例えば 64 ビットブロック長の暗号を CTR モードで利用した場合については、鍵を変更せずに 2^{32} ブロックすなわち 32GB 以上のデータを処理すると高い確率で無作為に選んだビット列と区別できることが知られている。さらに最近の研究 [5] によると具体的にビット列を導出できることも明らかになってきた。逆に、64 ビットブロック暗号を CTR モードで利用したとしても、ひとつの鍵で処理するデータ量が十分に小さければ、無作為に選んだビット列と区別できる確率が十分に小さいため、そのリスクを許容できる場合は効率的な利用法となり得るだろう。また、標準的ではないが CTR モードの代わりに CENC モード [6] や Abdalla-Bellare の

方法 [7] を利用することによりリスクを減らしたり回避したりできることもある。さらに、利用プロトコルもしくはシステム中で関連鍵攻撃が起きないような鍵管理がされている場合は、関連鍵攻撃耐性のない方式を使うことにより、効率をあげることが出来得る。

ブロック長に関する安全性指標については、ここにあげた通り、限界がかなりのところまで知られている。しかし、その他の安全性に関する性能指標については残念ながら分かっていないことが多い。例えば選択平文攻撃は出来ないが既知平文攻撃は想定の必要があるといった場合には明らかとなっていないことが殆んどである。暗号技術の安全性について「どんな攻撃に対しても何も起きない」といった「最強」の安全性についての評価の研究は進んでいるが、一部の軽量暗号で達成しようとしているような条件付きの安全性については研究結果が少なく、あまり明らかになっていないというのが実情である。電子政府推奨暗号や推奨候補暗号を利用したとしてもリスクなしでの運用は困難であり、軽量暗号の利用でも、利用に応じたリスクを考慮しながらの運用が必要である。また、軽量暗号といっても、全てにおいて電子政府推奨暗号や推奨候補暗号より劣っているわけではない。64 ビットブロック長なら、それに応じた安全性、関連鍵攻撃耐性を考慮しないなら、それに応じた安全性が達成されているので、必要な安全性とリスクを考慮した軽量暗号の利用が求められる。

1.4 今後の活動方針に対する提言

軽量暗号 WG では、2015 年度以降の軽量暗号に関する CRYPTREC での活動方針として、以下のような案 (A)(B)(C) を検討してきた（図 1.1 参照)。

それぞれの活動の目的と意義をまとめると下記のようになる。

(A)「暗号技術ガイドライン（軽量暗号の最新動向)」の発行
軽量暗号の最新技術動向をまとめた技術レポートであり、軽量暗号に関する情報や専門的知見を得るのに活用されることを目的とする。

(B)「暗号技術ガイドライン（軽量暗号の詳細評価)」の発行
代表的な軽量暗号アルゴリズムの安全性及び実装性能を統一的に評価した技術レポートであり、ユーザが軽量暗号アルゴリズムを選択・利用する際の技術的判断材料として活用できることを目的とする。これにより、軽量暗号の利用が促進されたり、軽量暗号に関する第三者評価レポートとして国際標準化等への寄書として活用されることが期待できる。

(C) 軽量暗号に関する技術公募の実施
CRYPTREC 暗号リストへの掲載を視野に、軽量暗号の公募・詳細評価を行い、選定を行う。これにより、軽量暗号が CRYPTREC 暗号リストへ新技術として追加され、電子政府システム等で最適な方式を選択でき、容易に調達できるようになることが期待される。

■今後の活動方針　軽量暗号は、特定の性能指標において既存技術と比べて優位性を持ち、M2M, IoT, CPS といった次世代のネットワークサービスを構築する上で有効なセキュリティ技術と期待される一方、電子政府推奨暗号リスト掲載の暗号技術ほど高い安全性を保証していない方式も存在しており、利用において留意すべき点がある。よって、軽量暗号を選択・利用する際の技術的判断の一助となり、今後の利用促進をはかることを目的として暗号技術ガイドラインを発行するのが有益と考えられる。

軽量暗号に関連する技術分野は多岐にわたり、分野ごとに研究開発の状況が異なる。ガイドライン作成にあたっては、詳細評価が望ましい分野や現時点では既存文献のサーベイで十分な分野など、各分野の状況を精査した上で、(A)

6

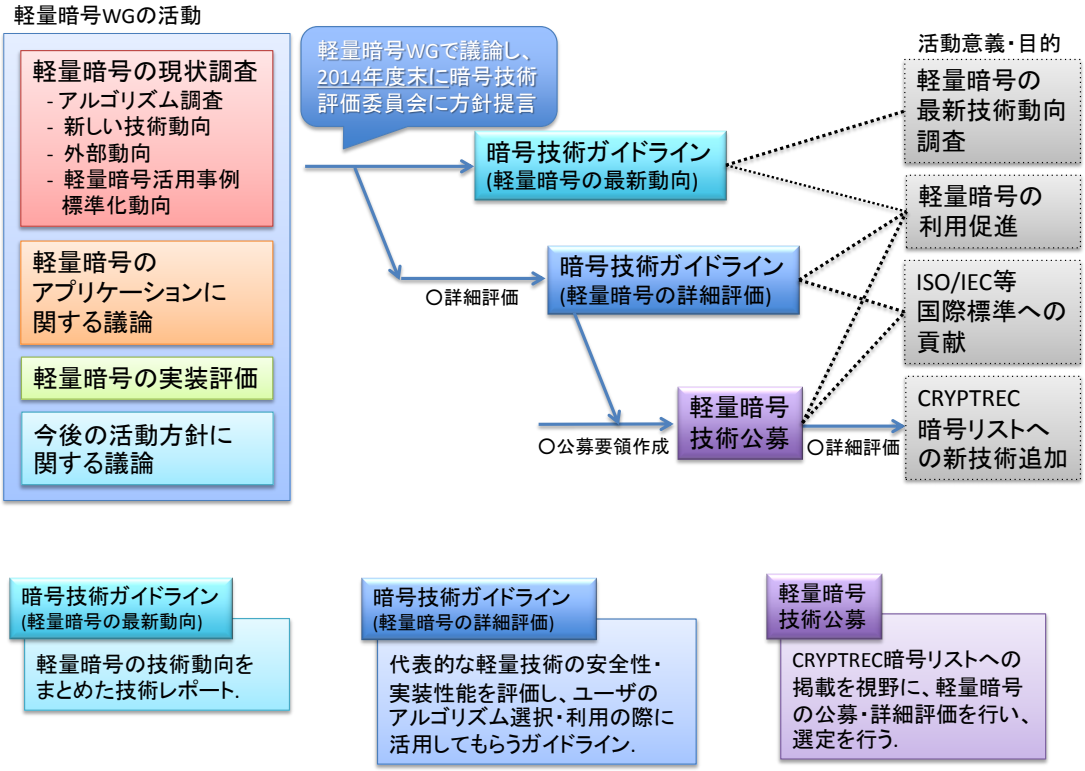


図 1.1　今後の活動方針案

と (B) のハイブリッド案でまとめるのが妥当と考えられる。詳細評価を行う技術分類は、新規評価の必要性（既存文献で十分な評価結果が得られるかどうか)、当該技術分野における我が国の技術の将来性、当該技術分野の現時点での注目度・重要度、評価結果から期待される学術的貢献等を鑑みて決定することが望ましい。

軽量暗号は、現時点では直ちに (C) の技術公募を行う段階ではないと考えるが、今後、IoT などの次世代ネットワークサービスで活用される可能性があることから、本 WG での検討が、長期的には電子政府システムの安全性向上にも資することが期待される。

参考文献

[1] STMicroelectronics, “CMP annual users meeting,” `http://cmp.imag.fr/aboutus/slides/Slides2013/05_ST_2013.pdf`, 2013.

[2] Mitsuo Usami, Hisao Tanabe, Akira Sato, Isao Sakama, Yukio Maki, Toshiaki Iwamatsu, Takashi Ipposhi and YasuoMiroslav Inoue, “A 0.05 × 0.05 mm^2 RFID Chip with Easily Scaled-Down ID-Memory,” ISSCC 2007, Digest of Technical Papers, pp. 482-483, 2007.

[3] Miroslab Kneevi, Ventzislav Nikov, and Peter Rombouts, “Low-Latency Encryption – Is “Lightweight= Light+ Wait”?” CHES 2012, pp. 426-446, 2012.

[4] Mitsuru Matsui and Yumiko Murakami, “Minimalism of Software Implementation - Extensive Performance Analysis of Symmetric Primitives on the RL78 Microcontroller,” FSE 2013, pp. 393-409, 2013.

[5] David A. McGrew, ”Impossible plaintext cryptanalysis and probable-plaintext collision attacks of 64-bit block cipher modes,” preproceedings of FSE 2013, Session 4-1.

[6] Tetsu Iwata, “New Blockcipher Modes of Operation with Beyond the Birthday Bound Security”, FSE 2006, pp.310-327, 2006.

[7] Michel Abdalla, Mihir Bellare, “Increasing the Lifetime of a Key: A Comparative Analysis of the Security of Re-keying Techniques,” ASIACRYPT 2000, pp. 546-559, 2000.

8

第 2 章

軽量暗号に関する現状調査: 軽量暗号アルゴリズム

2.1　軽量暗号に関する現状調査の概要

2013 年度および 2014 年度、軽量暗号 WG では、軽量暗号技術において、産業上のニーズがあり、具体的な暗号アルゴリズムの設計、安全性評価、実装評価が学会等で発表されている技術分類について、現状調査 (サーベイ) を行った。また、軽量暗号技術に関わる新しい技術動向や関連する外部動向についての調査や、軽量暗号の活用事例および標準化動向も行った。

これらの軽量暗号に関する現状調査を 2 章および 3 章にまとめる。

2 章の執筆担当者は下記の通りである。

第 2 章　軽量暗号アルゴリズム	
2.2 章　軽量ブロック暗号	青木委員、渋谷委員
2.3 章　軽量ストリーム暗号	渡辺委員
2.4 章　軽量ハッシュ関数	三宅委員
2.5 章　軽量メッセージ認証コード	渡辺委員
2.6 章　認証暗号	峯松委員、鈴木委員

2.2 軽量ブロック暗号

2.2.1 軽量ブロック暗号の安全性

本章では軽量暗号に分類されるブロック暗号の安全性に関する調査報告を行なう。軽量暗号に求められる安全性は暗号研究者の間でさえも合意されていない。もともと「『軽量』暗号」の名前の通り、安全性ではなく実装性能用件から始まった研究対象であり、「軽量とはいえ通常の暗号と同等の安全性が必要」という意見や、「通常利用しないような用途の安全性を犠牲にして軽量化を行なう」、また「通常利用しない用途の安全性は当然考慮せず、さらに通常使う用途に対しても長期間の安全性を保証せず、ぎりぎりを狙う」といった方式まである。従って、軽量暗号の利用に際しては、設計指針としてどこまでの安全性を考慮しているのかを理解して利用することが重要である。つまり従来型のブロック暗号の安全性と異なる部分が利用にあたって重要であることから本章では通常目的のブロック暗号に求められる安全性を調査する。

そもそも「何が『軽量ブロック暗号』か」という問に対して、軽量暗号という名前自体 buzz word と化しており難しい。広い意味で「軽量ブロック暗号」とされるものは [1] に詳しくあげられているが、AES など従来型のブロック暗号も含まれている。AES は電子政府推奨暗号であり、さらに事実上の世界標準であることから、本章では原則 AES より「軽量」なものを「軽量暗号」とした。軽量暗号の標準としては既に ISO/IEC で定められていることから ISO/IEC 29192 から中心に調査対象方式を選び、その他、共通鍵暗号の研究者の多くが「軽量」として引用している方式を調査対象とした[*1]。ここで、TDES, Camellia, CLEFIA については、平成 25 年に公表された CRYPTREC 暗号リストに掲載されており、安全性が十分に確認されている方式である。また、その後、本報告作成までの 2 年間の間に大きな問題は報告されていないので本章では調査対象外とする。

なお本章では、純粋にアルゴリズムそのものについての攻撃に対する安全性のみの調査を行ない、サイドチャネル攻撃や故障利用攻撃などは含めないこととする。また、秘密鍵の全数探索を高速化する手法、特に biclique を利用した中間一致攻撃的な手法 [4] がいくつかの暗号に対して提案されているが、効果は限定的であり、暗号の脆弱性として認められるのかどうかについても暗号研究者間で合意が得られていないのでこれも取り扱わないこととする。

表 2.1　軽量ブロック暗号の安全性評価

名称	提案文献	ブロック長	鍵長	仕様段数	攻撃可能段数	備考
LBlock	[23]	64	80	32	23	[6]
LED	[12]	64	64 ∼ 128	8/12	3/8	LED-64 と LED-128 に対応 [11]
Piccolo	[6]	64	80/128	25/31	9/11	whitening 鍵あり [17]
PRINCE	[2]	64	128	12	8	[9]
PRESENT	[5]	64	80/128	31	25(26)	26 段攻撃は全平文 [10]
PRINTcipher	[13]	48/96	80/160	48/96	48/96	[14] は弱鍵攻撃、[15] は関連鍵攻撃
TWINE	[19, 20]	64	80/128	36	23/25	[24, 21, 3]

[*1] 近年提案された Simon と Speck[8] については軽量暗号とみなされることが多い。提案論文そのものでは安全性評価が行なわれていないことから、解析論文が次々と出ている状態である。さらに、これらの方式は、パラメータが非常に多く、解析結果もそれぞれのパラメータに対して多数存在し、ここで最新情報を載せてもすぐに更新される可能性が高いことから今回は掲載を見送った。なお、現在 (2014 年 12 月) のところ、推奨パラメータでは破れていない。

参考文献

[1] Alex Biryukov and and Léo Perrin. State of the Art in Lightweight Cryptography. `http://cryptolux.org/index.php/Lightweight_Cryptography`, 2014.

[2] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.

[3] Alex Biryukov, Patrick Derbez, and Léo Perrin. Differential Analysis and Meet-in-the-Middle Attack against Round-Reduced TWINE. *preproceedings of Fast Software Encryption Workshop 2015 (FSE 2015)*, 2015.

[4] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology — ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Computer Science*, pages 344–371. Springer-Verlag, Berlin, Heidelberg, 2011.

[5] Andrey Bogdanov, Lars Ramkilde Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and Charlotte H. Vikkelsoe. PRESENT: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems — CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer-Verlag, Berlin, Heidelberg, New York, 2007.

[6] Christina Boura, María Naya-Plasencia, and Valentin Suder. Scrutinizing and Improving Impossible Differential Attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In Palash Sarkar and Tetsu Iwata editors, *Advances in Cryptology — ASIACRYPT 2014, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 179–199. Springer-Verlag, Berlin, Heidelberg, 2014.

[7] Özkan Boztas, Ferhat Karakoç, and Mustafa Çoban. Multidimensional Meet-in-the-Middle Attacks on Reduced-Round TWINE-128. LightSEC 2013.

[8] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The Simon and Speck Families of Lightweight Block Ciphers. IACR Cryptology ePrint Archive 2013/404.

[9] Anne Canteaut, María Naya-Plasencia, and Bastien Vayssière. Sieve-in-the-middle: Improved MITM attacks. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology — CRYPTO 2013, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 222–240, Berlin, Heidelberg, 2013. Springer-Verlag.

[10] Joo Yeon Cho. Linear cryptanalysis of reduced-round PRESENT. In Josef Pieprzyk, editor, *Topics in Cryptology - CT-RSA 2008: The Cryptographers' Track at the RSA Conference 2010*, volume 5985 of *Lecture*

Notes in Computer Science, pages 302–317, Berlin, Heidelberg, New York, 2010. Springer-Verlag.

[11] Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Key recovery attacks on 3-round Even-Mansour, 8-step LED-128, and full AES^2. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology — ASIACRYPT 2013, Part I*, volume 8269 of *Lecture Notes in Computer Science*, pages 337–356. Springer-Verlag, Berlin, Heidelberg, 2013.

[12] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer-Verlag, Berlin, Heidelberg, New York, 2011.

[13] Lars Ramkilde Knudsen, Gregor Leander, Axel Poschmann, and Matthew J. B. Robshaw. PRINTcipher: A block cipher for IC-printing. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems — CHES 2010*, volume 6225 of *Lecture Notes in Computer Science*, pages 16–32. Springer-Verlag, Berlin, Heidelberg, New York, 2010.

[14] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. A cryptanalysis of PRINTcipher: The invariant subspace attack. In Phillip Rogaway, editor, *Advances in Cryptology — CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 206–221, Berlin, Heidelberg, 2011. Springer-Verlag.

[15] Yuseop Lee, Kitae Jeong, Changhoon Lee, Jaechul Sung, and Seokhie Hong. Related-key cryptanalysis on the full PRINTcipher suitable for IC-printing. *International Journal of Distributed Sensor Networks*, 2014(Article ID 389476), 2014.

[16] David A. McGrew. Impossible plaintext cryptanalysis and probable-plaintext collision attacks of 64-bit block cipher modes. In Shiho Moriai, editor, *preproceedings of Fast Software Encryption Workshop 2013 (FSE 2013)*, Singapore, 2013.

[17] 芝山直喜, 金子敏信. Piccolo の新しい高階差分特性. 信学技報 ISEC2014-34, 2014.

[18] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An ultra-lightweight blockcipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 342–357. Springer-Verlag, Berlin, Heidelberg, New York, 2011.

[19] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE: A lightweight, versatile block cipher. In Gregor Leander and François-Xavier Standaert, editors, *ECRYPT Workshop on Lightweight Cryptography*, pages 146–169. ECRYPT II, 2011.

[20] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE : A lightweight block cipher for multiple platforms. In Lars Ramkilde Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography, 19th International Workshop, SAC 2012, Windsor, Ontario, Canada, August 15-16, 2012, Revised Selected Papers*, volume 7707 of *Lecture Notes in Computer Science*, pages 339–354, Berlin, Heidelberg, 2013. Springer-Verlag.

[21] Yanfeng Wang and Wenling Wu. Improved multidimensional zero-correlation linear cryptanalysis and applications to LBlock and TWINE. *Information Security and Privacy – 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings*, pages 1–16. 2014. Springer-Verlag.

[22] Long Wen, Meiqin Wang, Andrey Bogdanov, and Huaifeng Chen. Note of Multidimensional MITM Attack

on 25-Round TWINE-128. IACR Cryptology ePrint Archive 2014/425.

[23] Wenling Wu and Lei Zhang. LBlock: A Lightweight Block Cipher. In Javier Lopez and Gene Tsudik, *Applied Cryptography and Network Security — 9th International Conference, ACNS 2011, Nerja, Spain, June 7-10, 2011. Proceedings*, volume 2011 of *Lecture Notes in Computer Science*, pages 327–344. Berlin, Heidelberg, 2012. Springer-Verlag.

[24] Xuexin Zheng and Keting Jia. Impossible Differential Attack on Reduced-Round TWINE. *Information Security and Cryptology – ICISC 2013*, volume 8565 of *Lecture Notes in Computer Science*, pages 123–143, 2014. Springer-Verlag.

2.2.2 軽量ブロック暗号の実装性能

本章では、軽量暗号技術の現状調査として、主要な軽量ブロック暗号アルゴリズム、および CRYPTREC 暗号リストの電子政府推奨暗号リストに含まれるブロック暗号アルゴリズムの実装性能 (ハードウェア、ソフトウェア) 調査結果をまとめる。

2.2.2.1 調査対象

調査対象とした軽量ブロック暗号アルゴリズムは、ISO/IEC 29192 軽量暗号のパート 2 ブロック暗号に記載されているブロック暗号 (PRESENT、CLEFIA)、および主要国際学会で発表されており、現段階で有力な攻撃法が発見されておらず、かつ十分な実装性能を持つと考えられるアルゴリズム (LED、Piccolo、TWINE) とした。また、参考として、CRYPTREC 暗号リストの “電子政府推奨暗号リスト” に含まれるブロック暗号 (3-key Triple DES、AES、Camellia) の実装性能も調査した。さらに、類似の実装特性を持つ低レイテンシ暗号 PRINCE の実装性能も調査した。表 2.2 にこれら調査対象アルゴリズムをまとめる。実装性能の調査を行う論文としては、十分信頼が置けるデータが得られることを考慮し、各対象アルゴリズムの提案論文、および主要国際会議で発表された論文を中心に調査を行った。調査結果については、様々な資料から得られた評価値をできる限り公平になるように並べた。しかしながら、全ての評価が同じ環境で行われているわけではなく、評価環境や実装者によって評価値が変化する可能性があるため、本調査の数値は参考程度である点に注意されたい。

表 2.2 調査対象ブロック暗号アルゴリズム基本情報

Algorithm	Block size [bit]	Key size [bit]	# rounds	Structure	Ref.
3-Key Triple DES	64	168	48	Feistel	電子政府推奨暗号
AES	128	128/192/256	10/12/14	SPN	電子政府推奨暗号
Camellia	128	128/192/256	18/24/24	Feistel	電子政府推奨暗号
PRESENT	64	80/128	31	SPN	ISO/IEC29192-2
CLEFIA	128	128/192/256	18/22/26	GFN	ISO/IEC29192-2
LED	64	64/65 128	32/48	SPN	[11]
Piccolo	64	80/128	25/31	GFN	[28]
TWINE	64	80/128	36	GFN	[30]
PRINCE	64	128	12	SPN	[6]

2.2.2.2 ハードウェア実装性能調査

ハードウェア実装性能調査としては、十分な評価が行われていると考えられる ASIC での実装性能評価を調査した。実装性能の評価指標は、大別すると、主に自身で電源を持たないような機器 (passive device) 向けの指標として消費電力 (Power)、自身で電源を持つような機器 (active device) 向けの指標として消費電力量 (Energy) の 2 つがある。このうち、消費電力 (Power) における効率を示す指標としてはゲート規模がよく知られている。一方、消費電力量 (Energy)

の効率を示す指標としては、((ゲート規模)×(1-block 処理に必要なサイクル数)/(ブロックサイズ)) によって計算される energy per bit や、(((1-cycle で処理するビット数)×10^9)/(ゲート規模)^2) によって計算される FOM(Figure of Merit) が知られている。これらの調査結果を表 2.3-2.5 にまとめる。表中、Mode は暗号化関数のみを実装している場合は Enc と記載し、暗号化関数、復号関数をともに実装している場合は Enc/Dec と記載している。また、Area の評価として用いている GE は gate equivalent の略であり、ゲート規模を表す。Cycles/block は 1-block の演算に必要なサイクル数を表し、Throughput は、100[kHz] での Throughput のみを調査している。また、表中 LED* は LED の推定値による評価結果を示している。

表 2.3　128 bit ブロック暗号のハードウェア実装性能

Algorithm	Key size [bit]	Mode	Area [GE]	Cycles/ block	Throughput @100kHz [kbps]	Tech. [μm]	Ref.
AES	128	Enc/Dec	3,400	1,032/1,165	12.4/11.0	0.35	[10]
		Enc	2,400	226	56.6	0.18	[20]
		Enc/Dec	12,454	11	1,163.6	0.13	[27]
		Enc/Dec	5,398	54	237.0	0.13	
		Enc	3,100	160	80.0	0.13	[12]
Camellia	128	Enc/Dec	6,511	44	290.9	0.13	[27]
		Enc/Dec	6,264	44	290.9	0.18	
CLEFIA	128	Enc	2,488	328	39.0	0.13	[2]
		Enc/Dec	2,604	328/320	39.0/40.0	0.13	
		Enc	2,678	176	72.7	0.13	
		Enc/Dec	4,950	36	355.6	0.09	[29]
		Enc/Dec	5,979	18	711.1	0.09	

2.2.2.3　ソフトウェア実装性能調査

ソフトウェア実装性能調査として、ハイエンド CPU、およびローエンド CPU による実装評価の調査を行った。結果を表 2.6-2.8 にまとめる。ハイエンド CPU では実行速度として、Cycles/byte (1-byte の演算に必要なサイクル数) を調査し、ローエンド CPU では、Cycles/byte、および ROM、RAM 使用量をそれぞれ調査した。表 2.6 における Type は実装手法を表しており、それぞれ、Table による表引きを主に使用した実装を Table、VPI(Vector Permute Instruction) を利用した実装を VPI、bitslice 実装を Bitslice と記述している。Bitslice 実装における block 数の記述は並列実行ブロック数を表しており、例えば 8-block と記述があるものは、8-block 並列実行の bitslice 実装を表している。また、TWINE の実装手法における Single は通常の 1-block を実行する実装手法、Double は 2-block 並列に実行する実装手法を表す。

2.2.2.4　まとめ

本章では、軽量暗号技術の現状調査として、128-bit ブロック暗号アルゴリズム AES、Camellia、CLEFIA、および 64-bit ブロック暗号アルゴリズム 3-Key Triple DES、LED、Piccolo、TWINE、PRINCE の軽量暗号用途でのハー

ドウェア、ソフトウェア実装性能を公知の論文から調査した結果をまとめた。

表 2.4　64-bit ブロック暗号のハードウェア実装性能 (flexible-key setting)

Algorithm	Key size [bit]	Mode	Area [GE]	Cycles/ block	Throughput @100kHz [kbps]	Tech. [μm]	Ref.
Triple-DES	168	Enc/Dec	5,504	48	133.3	0.13	[27]
PRESENT	80	Enc	1,000	563	11.4	0.35	[26]
			1,570	32	200.0	0.18	[5]
	128	Enc	2,587	63	101.6	0.35	[26]
			2,681	39	164.1	0.35	[26]
			1,391	559	11.4	0.18	[24]
			1,886	32	200.0	0.18	[5]
LED	64	Enc	966	1,248	5.1	0.18	[11]
	128	Enc	1,265	1,872	3.4		
LED* (推定値)	64	Enc	2,695	32	200.0	0.18	[1]
	80	Enc	1,040	1,872	3.4	0.18	[11]
		Enc	2,780	48	133.3	0.18	[1]
	96	Enc	1,116	1,872	3.4	0.18	[11]
		Enc	2,866	48	133.3	0.18	[1]
	128	Enc	3,036	48	133.3	0.18	[1]
Piccolo	80	Enc	1,048	432	14.8	0.13	[14, 28]
		Enc/Dec	1,109	432	14.8		
		Enc	1,499	27	237.0		
		Enc/Dec	1,638	27	237.0		
	128	Enc	1,338	528	12.1		
		Enc/Dec	1,397	528	12.1		
		Enc	1,776	33	193.9		
		Enc/Dec	1,942	33	193.9		
TWINE	80	Enc	1,503	36	177.8	0.09	[30]
		Enc/Dec	1,799	36	177.8		
		Enc	1,011	393	16.3		
	128	Enc	1,866	36	177.8		
		Enc/Dec	2,285	36	177.8		
PRINCE	128	Enc/Dec	3,491	12	533.3	0.13	[6]
			2,953	12	533.3	0.13	[3]
			8,577	1	6,400		

16

表 2.5　64-bit ブロック暗号のハードウェア実装性能 (fixed-key setting)

Algorithm	Key size [bit]	Mode	Area [GE]	Cycles/ block	Throughput @100kHz [kbps]	Tech. [μm]	Ref.
LED	64	Enc	688	1,280	5.0	0.18	[11]
	128	Enc	700	1,872	3.4		
LED* (推定値)	64	Enc	2,354	32	200.0	0.18	[1]
	80	Enc	690	1,872	3.4	0.18	[11]
			2,354	48	133.3	0.18	[1]
	96	Enc	695	1,872	3.4	0.18	[11]
			2,354	48	133.3	0.18	[1]
	128	Enc	2,354	48	133.3	0.18	[1]
Piccolo	80	Enc	616	432	14.8	0.13	[14, 28]
		Enc/Dec	675	432	14.8		
		Enc	1,051	27	237.0		
		Enc/Dec	1,199	27	237.0		
	128	Enc	654	528	12.1		
		Enc/Dec	721	528	12.1		
		Enc	1,083	33	193.9		
		Enc/Dec	1,249	33	193.9		

表 2.6　128-bit ブロック暗号 (AES、Camellia) のソフトウェア実装性能 (ハイエンド CPU)

Algorithm	Key size [bit]	Type	Cycles/byte	Platform	Ref.
AES	128	VPI (Enc/Dec)	6.66/9.12	Core i5 U560	[30]
			7.42/9.44	Core i7 2600S	
			10.28/12.37	Core i3 2120	
			14.72/17.82	Xeon E5620	
			12.16/14.39	Core2Quad Q9550	
			22.04/25.82	Core2Duo E6850	
		Table (Enc/Dec)	14.26/19.27	Core i5 U560	
			14.04/21.17	Core i7 2600S	
			19.03/28.68	Core i3 2120	
			31.60/42.69	Xeon E5620	
			22.74/30.94	Core2Quad Q9550	
			22.43/30.76	Core2Duo E6850	
		Bitslice (8-block)	9.32	Core2Quad Q6600	[16]
			7.59	Core2Quad Q9550	
			6.92	Core i7 920	
	128	Bitslice (1/2/16-block)	10.7/7.8/5.4	PowerPC G4	[13]
	192		12.8/9.3/6.7		
	256		14.9/10.8/7.9		
Camellia	128	Bitslice (128-block)	9.19	Core2Duo E6400	[19]

表 2.7　64-bit ブロック暗号のソフトウェア実装性能 (ハイエンド CPU)

Algorithm	Key size [bit]	Type	Cycles/byte	Platform	Ref.
PRESENT	80/128	Bitslice (8/16/32-blk)	8.46/6.52/4.73	Xeon E3-1280	[17]
			10.88/7.26/5.79	Core i7 870	
			13.55/10.98/7.55	Xeon E5410	
	80	Table/VPI/Bitslice	72.6/35.0/17.4	Core i3 2367M	[4]
			65.7/42.1/20.7	Xeon X5650	
			59.5/42.3/21.0	Core2Duo P8600	
	128	Table/VPI/Bitslice	72.5/35.0/18.9	Core i3 2367M	
			65.7/42.1/24.1	Xeon X5650	
			59.5/42.4/24.1	Core2Duo P8600	
LED	64	Table/VPI/Bitslice	76.0/36.0/12.2	Core i3 2367M	[4]
			70.9/48.1/13.1	Xeon X5650	
			62.8/47.4/14.2	Core2Duo P8600	
	128	Table/VPI/Bitslice	113.3/54.6/17.6	Core i3 2367M	
			105.9/67.4/19.0	Xeon X5650	
			93.5/68.7/20.2	Core2Duo P8600	
Piccolo	80	Bitslice (16-blk)	4.57	Xeon E3-1280	[17]
			5.69	Core i7 870	
			6.85	Xeon E5410	
	128		5.52	Xeon E3-1280	
			6.80	Core i7 870	
			8.23	Xeon E5410	
	80	Table/VPI/Bitslice	83.9/33.3/9.2	Core i3 2367M	[4]
			71.0/37.4/9.7	Xeon X5650	
			67.1/38.3/10.7	Core2Duo P8600	
	128		103.6/41.6/10.9	Core i3 2367M	
			87.5/47.4/12.5	Xeon X5650	
			83.6/47.2/13.0	Core2Duo P8600	
TWINE	80/128	Single/Double	9.47/4.77	Core i5 U560	[30]
			11.10/5.55	Core i7 2600S	
			15.06/7.55	Core i3 2120	
			13.62/6.87	Xeon E5620	
			15.16/7.93	Core2Quad Q9550	
			26.85/14.85	Core2Duo E6850	

表 2.8　ブロック暗号のソフトウェア実装性能 (ローエンド CPU)

Algorithm	Block size [bit]	Key size [bit]	ROM [byte]	RAM [byte]	Cycles/ byte [Enc/Dec]	Platform	Ref.
AES	128	128	1,912	432	125/181	ATmega163	[7]
			1,659	33	287.5/4381	ATtiny45	[9]
			970	84	7,743/10,862	RL78	[18]
			1,989	64	3,917/5,911		
			2,380	64	3,865/5,706		
Camellia	128	128	1,020	78	39,357/152,023	RL78	[18]
			2,033	64	4,337/4,477		
			2,047	74	4,125/4,244		
CLEFIA	128	128	1,309	76	18,062/18,759	RL78	[18]
			2,026	64	7,768/7,799		
			2,040	86	6,208/6,740		
PRESENT	64	80	2,398	528	1,199/1,228	ATmega163	[24]
			1,000	18	1,412.5/1,700	ATtiny45	[9]
			936	0	1,340.4/1404.3	ATtiny45	[22]
			1,794	18	1090.1/-	ATtiny45	
			426	18	11,340.6/12,728.1	ATtiny45	
			512	62	61,634/60,834	RL78	[18]
			1,009	54	13,883/14,014		
			1,855	48	9,007/8,920		
TWINE	64	80	1,304	414	271/271	ATmega163	[30]
			728	335	2,350/2,337		
			792	191	2,350/2,337		
			2,294	386	163/163		
PRINCE	64	128	2,382	220	225.4	ATtiny85	[23]

参考文献

[1] The LED block cipher, December 2013. Available from `https://sites.google.com/site/ledblockcipher/hardware`.

[2] Toru Akishita and Harunaga Hiwatari. Very compact hardware implementations of the blockcipher CLEFIA. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, volume 7118 of *Lecture Notes in Computer Science*, pages 278–292. Springer, 2011.

[3] Lejla Batina, Amitabh Das, Baris Ege, Elif Bilge Kavun, Nele Mentens, Christof Paar, Ingrid Verbauwhede, and Tolga Yalçin. Dietary recommendations for lightweight block ciphers: Power, energy and area analysis of recently developed architectures. In Hutter and Schmidt [15], pages 103–112.

[4] Ryad Benadjila, Jian Guo, Victor Lomné, and Thomas Peyrin. Implementing lightweight block ciphers on x86 architectures. In Tanja Lange, Kristin E. Lauter, and Petr Lisonek, editors, *Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers*, volume 8282 of *Lecture Notes in Computer Science*, pages 324–351. Springer, 2013.

[5] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Paillier and Verbauwhede [21], pages 450–466.

[6] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.

[7] Joppe W. Bos, Dag Arne Osvik, and Deian Stefan. Fast implementations of AES on various platforms. *IACR Cryptology ePrint Archive*, 2009:501, 2009.

[8] Christophe Clavier and Kris Gaj, editors. *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September 6-9, 2009, Proceedings*, volume 5747 of *Lecture Notes in Computer Science*. Springer, 2009.

[9] Thomas Eisenbarth, Zheng Gong, Tim Güneysu, Stefan Heyse, Sebastiaan Indesteege, Stéphanie Kerckhof, François Koeune, Tomislav Nad, Thomas Plos, Francesco Regazzoni, François-Xavier Standaert, and Loïc van Oldeneel tot Oldenzeel. Compact implementation and performance evaluation of block ciphers in ATtiny devices. In Aikaterini Mitrokotsa and Serge Vaudenay, editors, *Progress in Cryptology - AFRICACRYPT*

2012 - 5th International Conference on Cryptology in Africa, Ifrance, Morocco, July 10-12, 2012. Proceedings, volume 7374 of *Lecture Notes in Computer Science*, pages 172–187. Springer, 2012.

[10] Martin Feldhofer, Johannes Wolkerstorfer, and Vincent Rijmen. AES implementation on a grain of sand. *Information Security, IEE Proceedings*, 152(1):13–20, 2005.

[11] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In Preneel and Takagi [25], pages 326–341.

[12] Panu Hämäläinen, Timo Alho, Marko Hännikäinen, and Timo D. Hämäläinen. Design and implementation of low-area and low-power AES encryption hardware core. In *Ninth Euromicro Conference on Digital System Design: Architectures, Methods and Tools (DSD 2006), 30 August - 1 September 2006, Dubrovnik, Croatia*, pages 577–583. IEEE Computer Society, 2006.

[13] Mike Hamburg. Accelerating AES with vector permute instructions. In Clavier and Gaj [8], pages 18–32.

[14] Harunaga Hiwatari, Kyoji Shibutani, Takanori Isobe, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Compact hardware implementations of the ultra-lightweight block cipher Piccolo. Proceedings of the ECRYPT Workshop on Lightweight Cryptography, 2011.

[15] Michael Hutter and Jörn-Marc Schmidt, editors. *Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers*, volume 8262 of *Lecture Notes in Computer Science*. Springer, 2013.

[16] Emilia Käsper and Peter Schwabe. Faster and timing-attack resistant AES-GCM. In Clavier and Gaj [8], pages 1–17.

[17] Seiichi Matsuda and Shiho Moriai. Lightweight cryptography for the cloud: Exploit the power of bitslice implementation. In Emmanuel Prouff and Patrick Schaumont, editors, *Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings*, volume 7428 of *Lecture Notes in Computer Science*, pages 408–425. Springer, 2012.

[18] Mitsuru Matsui and Yumiko Murakami. Minimalism of software implementation - extensive performance analysis of symmetric primitives on the RL78 microcontroller. In Shiho Moriai, editor, *Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers*, volume 8424 of *Lecture Notes in Computer Science*, pages 393–409. Springer, 2013.

[19] Mitsuru Matsui and Junko Nakajima. On the power of bitslice implementation on Intel Core2 processor. In Paillier and Verbauwhede [21], pages 121–134.

[20] Amir Moradi, Axel Poschmann, San Ling, Christof Paar, and Huaxiong Wang. Pushing the limits: A very compact and a threshold implementation of AES. In Kenneth G. Paterson, editor, *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, volume 6632 of *Lecture Notes in Computer Science*, pages 69–88. Springer, 2011.

[21] Pascal Paillier and Ingrid Verbauwhede, editors. *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*. Springer, 2007.

[22] Konstantinos Papagiannopoulos and Aram Verstegen. Speed and size-optimized implementations of the PRESENT cipher for tiny AVR devices. In Hutter and Schmidt [15], pages 161–175.

[23] Kostas Papapagiannopoulos. High throughput in slices: The case of PRESENT, PRINCE and KATAN64

ciphers. In Nitesh Saxena and Ahmad-Reza Sadeghi, editors, *Radio Frequency Identification: Security and Privacy Issues - 10th International Workshop, RFIDSec 2014, Oxford, UK, July 21-23, 2014, Revised Selected Papers*, volume 8651 of *Lecture Notes in Computer Science*, pages 137–155. Springer, 2014.

[24] Axel Poschmann. Lightweight cryptography - cryptographic engineering for a pervasive world. *IACR Cryptology ePrint Archive*, 2009:516, 2009.

[25] Bart Preneel and Tsuyoshi Takagi, editors. *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, volume 6917 of *Lecture Notes in Computer Science*. Springer, 2011.

[26] Carsten Rolfes, Axel Poschmann, Gregor Leander, and Christof Paar. Ultra-lightweight implementations for smart devices - security for 1000 gate equivalents. In Gilles Grimaud and François-Xavier Standaert, editors, *Smart Card Research and Advanced Applications, 8th IFIP WG 8.8/11.2 International Conference, CARDIS 2008, London, UK, September 8-11, 2008. Proceedings*, volume 5189 of *Lecture Notes in Computer Science*, pages 89–103. Springer, 2008.

[27] Akashi Satoh and Sumio Morioka. Hardware-focused performance comparison for the standard block ciphers AES, Camellia, and Triple-DES. In Colin Boyd and Wenbo Mao, editors, *Information Security, 6th International Conference, ISC 2003, Bristol, UK, October 1-3, 2003, Proceedings*, volume 2851 of *Lecture Notes in Computer Science*, pages 252–266. Springer, 2003.

[28] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: an ultra-lightweight blockcipher. In Preneel and Takagi [25], pages 342–357.

[29] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit blockcipher CLEFIA (extended abstract). In Alex Biryukov, editor, *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, volume 4593 of *Lecture Notes in Computer Science*, pages 181–195. Springer, 2007.

[30] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE: A lightweight block cipher for multiple platforms. In Lars R. Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, volume 7707 of *Lecture Notes in Computer Science*, pages 339–354. Springer, 2012.

2.3 軽量ストリーム暗号

本章では、軽量（lightweight）ストリーム暗号について報告する。

2.3.1 ECRYPT Stream Cipher project (eSTREAM)

eSTREAM は、EU における暗号技術研究 ECRYPT の一環として 2004〜2008 年に実施されたプロジェクトである。プロジェクトの中では、ストリーム暗号アルゴリズムの公募、および実装性能と安全性の両面から評価が行われた。eSTREAM プロジェクトでは、AES よりも性能が顕著であることを公募要件として求めており、ソフトウェア実装が高速であること (Profile I)、ハードウェア実装が軽量な暗号 (Profile II) のそれぞれの要件に特化したアルゴリズムを公募した。特にハードウェア実装の軽量性を追求した Profile II は、軽量暗号研究の流行を生んだ。CRYPTREC 軽量暗号 WG においては、low-latency 暗号など、処理速度の観点で軽量 (高速) と謳う暗号についても俎上に乗っているため、本報告では Profile I についても調査を行った。

表 2.9　eSTREAM Portfolio [1]

Profile 1 (ソフトウェア向け)	Profile 2 (ハードウェア向け)
HC-128 (128-bit), HC-256 (256-bit)	Grain-v1 (80-bit)
Rabbit (128-bit)	MICKEY 2.0 (80-bit, 128-bit)
Salsa20/12 (128-bit, 256-bit)	Trivium (80-bit)
SOSEMANUK (128-bit)	

当初の Portfolio では、Grain-128 が含まれていたが、2012 年の報告 [2] では Grain-128 を除外している。[2] によれば、Grain-128 は開発者自身がサポートしなくなったと報告されている。これは [11] で Grain-128 に弱鍵が存在する、およびセキュリティマージンが小さい、の 2 点が報告されたことが原因である。

2.3.1.1 Profile I (ソフトウェア実装が高速な暗号)

Profile I は PC、サーバ上で高速なソフトウェア向け暗号を指向しており、鍵長は 128 ビット以上である。eCRYPT II の 1 プロジェクトである VAMPIRE の成果である eBACS [9] で、さまざまな環境での処理速度を確認することができる。表 2.10 は、Intel Core i5（64 ビットモード）における評価結果である (詳細：Intel Core i5-2400S; 4 x 2495MHz; sandy, supercop-20120908)。

また、2009 TI Sitara AM3703 500MHz (ARM Cortex A8) 上での処理性能は表 2.11 のとおりである (詳細：armeabi (v7-A, Cortex A8); 2009 TI Sitara AM3703; 1 x 500MHz; h7silver, supercop-20130126)。

Profile I に属するアルゴリズムは、いずれも AES(AES-NI 不使用の場合) に比べて 3〜5 倍のスループットを実現している。AES 命令が実装されていない環境では利用に適するケースもある。現在、Salsa20 は TLS 用の暗号スイートとして提案が進められている [7]。

アルゴリズム構造は算術演算を用いるもの (Rabbit, Salsa20/12)、大きな内部状態を持ち、初期化に時間をかけるもの (HC-128, SOSEMANUK) の 2 系統に分かれる。後者のアルゴリズムは短いデータの処理には適していない。また、Profile I に属するアルゴリズムは、ハードウェア実装したときに論理規模が大きくなるケースが多いと考えられる。

HC-128, Rabbit は組み込み機器向けの SSL/TLS 実装 ChaSSL に実装されている [2]。また、Rabbit は

表 2.10　eSTREAM Portfolio Profile I アルゴリズムのソフトウェア実装性能（Intel Core i5）[8]

	処理速度 (cycle/B)				
	長いメッセージ	4096B	576B	64B	8B
HC-128	2.32	7.13	36.44	309.25	2472.00
Rabbit	4.41	4.58	5.41	13.06	80.00
Salsa20/12	2.40	2.44	2.70	4.94	56.50
SOSEMANUK	3.54	3.81	5.72	20.56	164.50
AES	11.33	11.41	11.78	15.75	77.50
KCipher-2(*)	4.01	4.22	5.50	17.45	111.51

CRYPTREC 電子政府推奨暗号との比較のため、KCipher-2 の処理性能を [14] に記載されている性能から見積もった。なお、[14] の評価環境は Intel Core2Duo である。

表 2.11　eSTREAM Portfolio Profile I アルゴリズムのソフトウェア実装性能（ARM Cortex A8）[8]

	処理速度 (cycle/B)				
	長いメッセージ	4096B	576B	64B	8B
Salsa20/12	5.52	5.84	8.14	28.50	264.75
AES	19.28	20.36	29.59	111.83	852.38

ISO/IEC 18033-4 [5] および RFC 4503 [7] に記載されている。

2.3.1.2　Profile II (ハードウェア実装規模／消費電力が小さい暗号)

Profile II は軽量なハードウェア実装向け暗号を指向しており、鍵長は 80 ビット以上である。軽量暗号の実装では、状態を保持するレジスタが論理回路の大半を占めることから、回路規模削減のために、Profile I に比べて短い鍵長を許容しているものと考えられる。鍵長 128 ビットレベルセキュリティを持つアルゴリズムに比べると、安全性が低く設定されているので、用途は限定されるべきである。

表 2.12 および図 2.1 は、文献 [10] から抜粋した Profile II (および AES) のハードウェア実装性能である。いずれのアルゴリズムも、論理規模、処理速度の両面で AES に比べて顕著な軽量性を実現している。軽量実装では、回路の動作周波数が低く抑えられているケースが多いと想定されるため、[10] では動作周波数を 10MHz, 100kHz に固定した場合の消費電力評価も行われている。消費電力はアルゴリズムによらず、論理規模に比例して増加する傾向が見られた。

2.3.2　ISO/IEC 29192-3

ISO/IEC JTC 1/SC 27 では、一般的な暗号アルゴリズムの標準を定めた ISO/IEC 18033 に加えて、軽量暗号の標準を ISO/IEC 29192 で定めている。ストリーム暗号は 2012 年に発行された Part 3 に収められており、eSTREAM Portfolio に掲載された Trivium (鍵長 80 ビット) と、CRYPTREC 推奨候補暗号に掲載された Enocoro (鍵長 80 ビットおよび 128 ビット) の 2 つのアルゴリズムが収録されている。Enocoro-80, Enocoro-128v2 のハードウェア実装性能を表 2.13 にまとめる。Trivium の実装性能については紹介済みなので割愛する。Enocoro の性能は eSTREAM Portfolio II に掲載のアルゴリズムと同程度である。消費電力に関する情報は見つかっていない。

表 2.12　eSTREAM Portfolio Profile II アルゴリズムのハードウェア実装性能 [10]

アルゴリズム	出力	最大動作周波数	スループット	論理規模	実装プロセス
	(bit/cycle)	(MHz)	(Mbps)	(kgate)	(μm)
Grain	1	724.6	724.6	1.3	0.13
	8	632.9	5063.2	2.2	
Trivium	1	327.9	327.9	2.6	
	8	471.7	3773.6	3.0	
Mickey 2.0	1	454.5	454.5	3.2	
Enocoro-80	8	274.7	2197.6	2.7	0.18
AES	2.37	131.2	311.0	5.4	0.11
	0.124	80.0	10.0	3.4	0.35

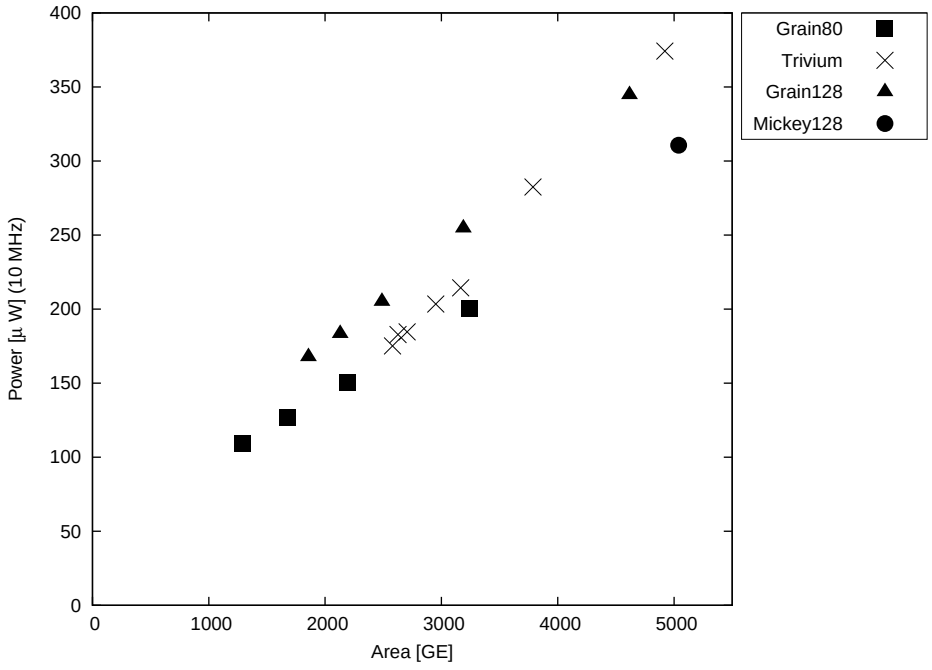


図 2.1　eSTREAM Portfolio Profile II アルゴリズムのハードウェア実装規模と消費電力 [10]

表 2.13　Enocoro のハードウェア実装性能

アルゴリズム	最大動作周波数 (MHz)	スループット (Mbps)	論理規模 (kgate)	実装プロセス (μm)
Enocoro-80 [12]	274.7	2197.6	2.7	0.18
Enocoro-128v2 [13]	440.0	3520.0	4.1	0.09

参考文献

[1] Steve Babbage, Christophe De Cannière, Anne Canteaut, Carlos Cid, Henri Gilbert, Thomas Johansson, Matthew Parker, Bart Preneel, Vincent Rijmen, and Matthew Robsha, “The eSTREAM Portfolio (rev. 1),” September 8, 2008.

[2] Carlos Cid and Matt Robshaw, The eSTREAM Portfolio in 2012, ECRYPT II, European Network of Excellence in Cryptology II, 2012.

[3] M. Robshaw and O. Billet, editors. New Stream Cipher Designs: The eSTREAM Finalists, LNCS 4986, pp. 267–293. Springer.

[4] M. Boesgaard, M. Vesterager, and E. Zenner. A Description of the Rabbit Stream Cipher Algorithm. Network Working Group, Request for Comments 4503. `http://tools.ietf.org/html/rfc4503`

[5] ISO/IEC 18033-4. Information technology – Security techniques – Encryption algorithms – Part 4: Stream ciphers. 2005.

[6] ISO/IEC 29192-3:2012, Information technology – Security techniques – Lightweight cryptography – Part 3: Stream ciphers, 2012.

[7] A Description of the Rabbit Stream Cipher Algorithm, RFC4503.

[8] “The Salsa20 Stream Cipher for Transport Layer Security,” draft-josefsson-salsa20-tls-04, November 26, 2013.

[9] eBASC: ECRYPT Benchmarking of Stream Ciphers

[10] T. Good and M. Benaissa, “Hardware performance of eStream phase-III stream cipher candidates,” SASC2008, Feb 2008.

[11] I. Dinur and A. Shamir. Breaking Grain-128 with Dynamic Cube Attacks. In A. Joux, editor, Proceedings of FSE 2011, LNCS 6733, pp. 167–187, Springer, 2011.

[12] Watanabe, D., Ideguchi, K., Kitahara, J., Muto, K., Furuichi, H., Kaneko, T. “Enocoro-80: A Hardware Oriented Stream Cipher”. Third International Conference on Availability, Reliability and Security, pp. 1294–1300, IEEE Press, New York, 2008.

[13] 日立製作所, 「ストリーム暗号 Enocoro 評価書」, 2010.

[14] KDDI 研究所, 「ストリーム暗号 KCipher-2」, CRYPTREC シンポジウム 2010, 2010.

2.4 軽量ハッシュ関数

本章では、軽量暗号技術の現状調査として、代表的なハッシュ関数の安全性と実装性能に関する調査結果を報告する。

2.4.1 調査対象アルゴリズム

調査対象とするハッシュ関数アルゴリズムは、軽量ハッシュ関数として主要国際会議で提案された PHOTON、SPONGENT、QUARK、および、CRYPTREC 暗号リストに含まれる SHA-2、SHA-3 として選定された Keccak の 5 方式とする。調査文献として、SHA-2、SHA-3 については NIST の SHA-3 Competition から、その他のアルゴリズムについては提案論文や主要国際会議の論文を中心に調査した。

2.4.2 安全性

アルゴリズムの構造に基づいた Generic attacks に対する安全性に関し、各アルゴリズムの preimage attack、2nd-preimage attack、collision attack に対する計算複雑度を表 2.14 に示す。Sponge 構造である Keccak、PHOTON、SPONGENT、QUARK については、“Parameter” の “n” は hash size を、“c” は capacity を、“r” は rate を表している。Merkle-Damgard 構造である SHA-1、SHA-2 に関しては、それぞれ hash size、internal state size、message block size を表している。アルゴリズム特有の性質を利用した攻撃手法については、SHA-2 や Keccak に対しては数多くの報告があるが（CRYPTREC 技術報告書 [1],[2] 参照)、それ以外のアルゴリズムに対してはまだ十分に議論されていないため安全性に欠陥がある可能性がある。

2.4.3 ハードウェア実装性能

ハードウェア実装性能に関する調査結果を表 2.15, 2.16 に纏める。これらの評価値は様々な文献から抽出したものであり、同一環境で評価されたものではないことに注意されたい。参考情報として AES ベースのハッシュ関数の実装性能（推定値を含む）についても記載した。表中の “Area” はゲート規模を、“Latency” は internal permutation P (または internal block-cipher E) のクロック数を、“FOM(Figure of Merit)” はエネルギー効率を表す指標（スループットとゲート規模の二乗の比）を、“Power” は平均消費電力を表している（動作周波数 100kHz での性能を示す)。この結果より、軽量ハッシュ関数と呼ばれるアルゴリズムは、回路規模を数 kGE 程度に収めることを優先し、スループットはあまり高くない設計のものが多いことが分かる。

2.4.4 ソフトウェア実装性能

ソフトウェア実装性能については、軽量暗号という観点から非力な CPU（Atmel AVR ATtiny45 8-bit RISC microcontroller）での性能を示した CARDIS2012 の発表論文 [12] を引用する（表 2.17)。

2.4.5 まとめ

CRYPTREC 暗号リストに記載の SHA-2 と SHA-3 として選定された Keccak、および軽量ハッシュ関数に分類される PHOTON、SPONGENT、QUARK について、提案論文を中心に安全性と実装性能について調査した。軽量ハッシュ関数は 64〜128 ビットセキュリティの安全性での実装性能を重視したものが多く、ハードウェア実装性能において

表 2.14　各アルゴリズムの安全性

Algorithm	Hash [bit]	Parameter[bit] n	c	r	Security[bit] Pre	2nd-Pre	Col	Source
SHA-1	160	160	160	512	160	160	80	[8]
SHA-256	256	256	256	512	256	256	128	[8]
Keccak-f[200]*1	128	200	128	72	128	128	64	[5]
Keccak-f[400]*1	160	400	256	144	160	160	80	[5]
PHOTON-80	80	80	80	20	64	40	40	[8]
PHOTON-128	128	128	128	16	112	64	64	[8]
PHOTON-160	160	160	160	36	124	80	80	[8]
PHOTON-224	224	224	224	32	192	112	112	[8]
PHOTON-256	256	256	256	32	224	128	128	[8]
SPONGENT-88	88	88	80	8	80	40	40	[10]
SPONGENT-128	128	128	128	8	120	64	64	[10]
SPONGENT-160	160	160	160	16	144	80	80	[10]
SPONGENT-224	224	224	224	16	208	112	112	[10]
SPONGENT-256	256	256	256	16	240	128	128	[10]
U-QUARK	128	136	128	8	128	64	64	[11]
D-QUARK	160	176	160	16	160	80	80	[11]
S-QUARK	224	256	224	32	224	112	112	[11]

SHA-2 と比較すると、回路規模の面で大きな優位性があるものの、速度面では必ずしも優れているわけではなく、レイテンシは勝るものもあるがスループットに関しては概ね劣っていることが分かった。以上の観点から、今回調査した軽量ハッシュ関数は、特に回路規模に制限があるデバイスや低レイテンシが要求されるアプリケーションでの利用が適していると考えられる。

*1 Keccak-f[] は置換関数であることに注意

表 2.15　ハードウェア実装性能

Algorithm	Area [GE]	Latency [clk]	Throughput [kbps]	FOM	Power [uW]	Proc. [nm]	Source
SHA-1	6,812	450	113.78	24.52	11.0	250	[3]
SHA-256	8,588	490	104.48	14.17	11.2	250	[4]
KECCAK-f[200]	2,520	900	8.00	12.60	5.60	130	[5]
	4,900	18	400.0	166.6	27.6	130	[5]
KECCAK-f[400]	5,090	1,000	14.40	5.56	11.5	130	[5]
	10,560	20	720.00	64.57	78.1	130	[5]
KECCAK-f[1600]	20,790	1,200	90.66	2.10	44.9	130	[5]
	47,630	24	4,533	19.98	315.1	130	[5]
AES-based DM scheme-128	>4,400	-	<12.4	-	-	-	[7]
AES-based Hirose scheme-256	>9,800	-	<12.4	-	-	-	[7]
PHOTON-80/20/16	865	708	2.82	37.73	1.59	180	[8]
	1,168	132	15.15	111.13	2.70	180	[8]
	1,067	708	2.82	24.77	14.0	45	[9]
	1,567	132	15.15	61.70	39.9	45	[9]
PHOTON-128/16/16	1,122	996	1.61	12.78	2.29	180	[8]
	1,708	156	10.26	35.15	3.45	180	[8]
	1,394	996	1.61	8.29	17.2	45	[9]
	2,172	156	10.26	21.75	49.6	45	[9]
PHOTON-160/36/36	1,396	1332	2.70	13.87	2.74	180	[8]
	2,117	180	20.00	44.64	4.35	180	[8]
	1,741	1332	2.70	8.91	19.4	45	[9]
	2,849	180	20.00	24.64	65.8	45	[9]
PHOTON-224/32/32	1,735	1716	1.86	6.19	4.01	180	[8]
	2,786	204	15.69	20.21	6.50	180	[8]
	2,142	1716	1.86	4.05	22.6	45	[9]
	3,586	204	15.69	12.20	78.8	45	[9]
PHOTON-256/32/32	2,177	996	3.21	6.78	4.55	180	[8]
	4,362	156	20.51	10.78	8.38	180	[8]
	2,675	996	3.21	4.49	51.6	45	[9]
	5,335	156	20.51	7.21	248.	45	[9]

30

表 2.16　ハードウェア実装性能 (続)

Algorithm	Area [GE]	Latency [clk]	Throughput [kbps]	FOM	Power [uW]	Proc. [nm]	Source
SPONGENT-88	738	990	0.81	14.9	1.57	130	[10]
	1,127	45	17.78	139	2.31	130	[10]
	869	990	0.81	10.7	16.5	45	[9]
	1,237	45	17.78	116	38.7	45	[9]
SPONGENT-128	1,060	2,380	0.34	3.03	2.20	130	[10]
	1,687	70	11.43	40.2	3.58	130	[10]
	1,257	2,380	0.34	2.15	21.1	45	[9]
	1,831	70	11.43	34.1	53.2	45	[9]
SPONGENT-160	1,329	3,960	0.40	2.26	2.85	130	[10]
	2,190	90	17.78	37.1	4.47	130	[10]
	1,572	3,960	0.40	1.62	24.6	45	[9]
	2,406	90	17.78	30.7	73.5	45	[9]
SPONGENT-224	1,728	7,200	0.22	0.7	3.73	130	[10]
	2,903	120	13.33	15.8	5.97	130	[10]
	2,070	7,200	0.22	0.5	31.4	45	[9]
	3,220	120	13.33	12.9	96.0	45	[9]
SPONGENT-256	1,950	9,520	0.17	0.45	4.21	130	[10]
	3,281	140	11.43	10.6	6.62	130	[10]
	2,323	9,520	0.17	0.32	34.2	45	[9]
	3,639	140	11.43	8.63	110.	45	[9]
U-QUARK	1,379	544	1.47	7.73	2.44	180	[11]
	2,392	68	11.76	20.6	4.07	180	[11]
	1,744	544	1.47	4.83	51.2	45	[9]
	3,215	68	11.76	11.4	89.4	45	[9]
D-QUARK	1,702	704	2.27	7.84	3.10	180	[11]
	2,819	88	18.18	22.9	4.76	180	[11]
	2,200	704	2.27	4.69	58.6	45	[9]
	3,695	88	18.18	13.3	87.7	45	[9]
S-QUARK	2,296	1,024	3.13	5.94	4.35	180	[11]
	4,640	64	50.0	23.2	8.39	180	[11]
	3,001	1,024	3.13	3.48	81.6	45	[9]
	6,155	64	50.0	13.2	146	45	[9]

表 2.17　ソフトウェア実装性能

Algorithm	Digest size [bits]	Code size [bytes]	RAM data [bytes]	RAM state & others [bytes]	RAM stack	Cycle count (8byte msg)	Cycle count (50byte msg)	Cycle count (100byte msg)	Cycle count (500byte msg)
SHA-256	256	1090	64	73	6	33,600	33,600	66,815	266,105
Keccak[r =40, c =160]	160	752	5	45	3	58,063	162,347	278,269	1,205,627
Keccak[r =144, c =256]	256	608	18	92	4	90,824	181,466	317,221	1,313,291
Keccak[r =1088, c =512]*	256	868	136	240	4	178,022	178,022	179,494	716,483
PHOTON-160/36/36	160	764	9	39	11	620,921	1,655,364	2,793,265	11,999,914
PHOTON-256/32/32	256	1,244	4	68	10	254,871	486,629	787,896	3,105,396
SPONGENT-160/160/80	160	598	10	60	6	795,294	2,783,241	4,771,186	20,674,746
SPONGENT-256/256/128	256	364	16	96	5	1,542,923	3,856,916	6,170,900	25,454,100
D-QUARK	176	974	2	42	5	631,871	1,516,685	2,570,035	10,996,835
S-QUARK	256	1106	4	60	5	708,783	1,417,611	2,339,023	9,427,023

参考文献

[1] 盛合志帆, ハッシュ関数の安全性に関する技術調査報告書, CRYPTREC 技術報告書 No.0213, http://www.cryptrec.go.jp/estimation.html#2004, 2004.

[2] 金子敏信, SHA-256/-384/-512 の評価報告, CRYPTREC 技術報告書 No.0503, http://www.cryptrec.go.jp/estimation.html#2005, 2005.

[3] Mooseop Kim and Jaecheol Ryou, Power Efficient Hardware Architecture of SHA-1 Algorithm for Trusted Mobile Computing. In Sihan Qing, Hideki Imai, and Guilin Wang, editors, *Information and Communications Security, 9th International Conference — ICICS 2007*, volume 4861 of *Lecture Notes in Computer Science*, pages 375-385, Springer-Verlag, Berlin, Heidelberg, 2007.

[4] Mooseop Kim, Jaecheol Ryou, and Sungik Jun, Efficient Hardware Architecture of SHA-256 Algorithm for Trusted Mobile Computing. In Moti Yung, Peng Liu, and Dongdai Lin, editors, *Information Security and Cryptology — ISC 2008*, volume 5487 of *Lecture Notes in Computer Science*, pages 240-252, Springer-Verlag, Berlin, Heidelberg, 2009.

[5] Elif Bilge Kavun and Tolga Yalcin, A Lightweight Implementation of Keccak Hash Function for Radio-Frequency Identification Applications, In Siddika Berna Ors Yalcin, editor, *Radio Frequency Identification: Security and Privacy Issues — RFIDsec 2010*, volume 6370 of *Lecture Notes in Computer Science*, pages 258-269, Springer-Verlag, Berlin, Heidelberg, 2010.

[6] Luca Henzen, Pietro Gendotti, Patrice Guillet, Enrico Pargaetzi, Martin Zoller, and Frank K. Gürkaynak, Developing a Hardware Evaluation Method for SHA-3 Candidates, In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems — CHES 2012*, volume 6225 of *Lecture Notes in Computer Science*, pages 248-263, Springer-Verlag, Berlin, Heidelberg, 2010.

[7] Andrey Bogdanov, Gregor Leander, Christof Paar, Axel Poschmann, Matt J. B. Robshaw, and Yannick Seurin, Hash Functions and RFID Tags: Mind the Gap, In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 283-299, Springer-Verlag, Berlin, Heidelberg, 2008.

[8] Jian Guo, Thomas Peyrin, and Axel Poschmann, The PHOTON Family of Lightweight Hash Functions, In Phillip Rogaway, editor, *Advances in Cryptology — CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 222-239, Springer-Verlag, Berlin, Heidelberg, 2011.

[9] Andrey Bogdanov, Miroslav Knežević, Gregor Leander, Deniz Toz, Kerem Varici, and Ingrid Verbauwhede, SPONGENT: The Design Space of Lightweight Cryptographic Hashing, volume 62, issue 10, pages 2041-2053, IEEE Transactions on Computers, 2013.

[10] Andrey Bogdanov, Miroslav Knežević, Gregor Leander, Deniz Toz, Kerem Varici, and Ingrid Verbauwhede,

SPONGENT: A Lightweight Hash Function, In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 312-325, Springer-Verlag, Berlin, Heidelberg, 2011.

[11] Jean-Philippe Aumasson, Luca Henzen, Willi Meier, Mara Naya-Plasencia, Quark: A Lightweight Hash, In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems — CHES 2010*, volume 6225 of *Lecture Notes in Computer Science*, pages 1-15, Springer-Verlag, Berlin, Heidelberg, 2010.

[12] Josep Balasch, Barış Ege, Thomas Eisenbarth, Benoit Gérard, Zheng Gong, Tim Güneysu, Stefan Heyse, Stéphanie Kerckhof, François Koeune, Thomas Plos, Thomas Pöppelmann, Francesco Regazzoni, François-Xavier Standaert, Gilles Van Assche, Ronny Van Keer, Loïc van Oldeneel tot Oldenzeel, and Ingo von Maurich, Compact Implementation and Performance Evaluation of Hash Functions in ATtiny Devices, In Stefan Mangard, editor, *Smart Card Research and Advanced Applications — CARDIS 2012*, volume 7771 of *Lecture Notes in Computer Science*, pages 158-172, Springer-Verlag, Berlin, Heidelberg, 2013.

2.5 軽量メッセージ認証コード

本章では、軽量なメッセージ認証コード (Message Authentication Code: MAC) に関する調査報告を行う。

MAC は、nonce 入力の有無によって、probabilistic MAC と deterministic MAC に分けられる。nonce 入力を持つ probabilistic MAC は、リプレイ攻撃に耐性を持つ。また、Wegman と Carter が提案した、universal hash function から MAC を構成する方法 [9] は nonce 入力が必須である。universal hash function をベースとする MAC は、代数的演算を用いることを特徴としてお、64 ビットレジスタ上での演算などを高速に行うことができる実装環境では優れた処理性能を発揮する。

一方、ブロック暗号から構成する CMAC [18] や、ハッシュ関数から構成する HMAC [2, 26] は deterministic MAC として定義される（nonce を prefix とすることで probabilistic MAC として用いることもできる)。このような、暗号プリミティブからモードとして MAC を定義する場合、実装環境に合わせて軽量な暗号プリミティブを使用することで、軽量な MAC となることが期待できる。

2.5.1 universal hash function を用いる構成法

Wegman と Carter により、ユニバーサルハッシュ関数 h から安全な MAC を構成できることが知られている [27]。Wegman-Carter 方式による MAC の構成は $MAC(m,k,r) = h(m,k) + b(r)$ で定義される。ただし、ここで $b(r)$ は one-time key である。主要なユニバーサルハッシュ関数はいずれも代数的な演算で構成されており、これらの演算が高速に実行できる環境では優れた処理性能を実現する。

まず、Wegman らが提案した多項式を用いる方式 (polynomial hashing) では、ユニバーサルハッシュ関数は $h(m,k) = \sum_i m_i k^i$ で定義される。polynomial hashing のアルゴリズム例として GMAC [13] や Poly1305 [3] がある。GMAC の演算は標数 2 の拡大体 GF(2^{128}) 上で、また、Poly1305 の演算は素体 $GF(2^{130}-5)$ 上で定義される。Saarinen は GMAC に弱鍵があることを指摘している [25]。また、Procter らは、この脆弱性が多項式の取り方に依らず存在すること、および任意の鍵を弱鍵と見做せることを示した [24]。しかし、Procter の攻撃では、弱鍵を検出する識別子のメッセージ長と、弱鍵の空間の大きさが等価であるため、現実的な脅威ではないと考えられる。

[15] によれば、Intel Haswell アーキテクチャ上では GMAC (GHASH) の漸近的な処理速度は 0.4 cycle/Byte である。また、表 2.18 は、[4] で提供されている Poly1305-AES の処理性能から抜粋したものである。

表 2.18　Poly1305-AES の処理速度 [4](単位：cycles/Byte)

	データ長			
	64	256	1024	long
Pentium III	16.3	6.9	5.1	4.4
Pentium 4	18.7	8.0	5.3	4.5
Athlon	13.1	5.7	3.7	3.2

また、Halevi と Krawczyk は内積を用いる方式 MMH を提案した [17]。MMH はメッセージ $m = \{m_1, \ldots, m_n\}$ と等長の鍵ストリーム $k = \{k_1, \ldots, k_n\}$ に対して $h(m,k) = \sum_i m_i \cdot k_i$ 定義される。UMAC [7] や Badger [8] は MMH と同じく内積方式であるが、MMH が有限体上の演算を用いて定義されているのに対して、ソフトウェア実装に適した $Z/2^wZ$ 上の演算を用いる点が異なる。

MMH 方式では、一般に鍵をメッセージと等長のビット列に伸長して内積を計算する。したがって、他の方式に比べて事前処理に要するコストが大きくなる。また、拡大鍵を保持するためのメモリ使用量が増大する傾向にあり、複数の相手と通信を行うようなケースでは、メモリを圧迫する可能性がある。[7] や [8] では、安全に拡大鍵を使い回す方法や、tree-hash との組み合わせにより拡大鍵の量を削減する方法が紹介されている。

表 2.19 は、[21] で報告されている UMAC の処理性能である。報告されている数値からの推定になるが、UMAC の性能には、少くとも鍵を伸長する事前処理は含まれていないと考えられる。

表 2.19　Pentium 4 上での UMAC の処理速度 [21](単位：cycles/Byte)

	データ長			
タグ長	64	256	1024	long
32	8.3	2.4	0.9	0.6
64	12.0	3.5	1.4	1.0
96	15.1	4.5	1.9	1.5

また、表 2.20 は [8] で報告されている、タグ長が 64 ビットの Badger の処理速度である。

表 2.20　Pentium III および Pentium 4 上での Badger の処理速度 [8]

	事前処理	メッセージ処理	最終処理
Pentium III	4,093 cycles	2.2 cycles/Byte	433 cycles
Pentium 4	5,854 cycles	1.3 cycles/Byte	800 cycles

上に挙げた方式の実装性能はいずれも、CPU が 64 ビットアーキテクチャやベクトル演算を利用可能な環境、もしくは多大な事前計算テーブルをメモリに展開できる環境において実現されたものであり、計算機能力が貧弱な環境には適していない可能性が高い。

2.5.2　暗号プリミティブを用いる構成法

暗号プリミティブから MAC を構成する方法として、ブロック暗号から構成する CMAC [18] や、ハッシュ関数から構成する HMAC [2, 26] がある。ISO/IEC 9797 [28, 29] には、CMAC や HMAC の他にも、CBC-MAC のバリエーションなどが規定されている。Bertoni らが [5] でスポンジ関数を提案して以降、置換をベースとする暗号機能の研究がさかんになった。MAC の構成法としては、secret-prefix 方式が一般的であり、Bertoni らにより、その安全性が証明されている [6]。多くの軽量ハッシュ関数はスポンジ関数から構成されているので、上記の secret-prefix 方式を用いることが可能である。スポンジ関数の secret-prefix 方式は最終処理が不要であるため、メッセージ長が短い場合には、HMAC に比べて処理時間が短いことが期待される。

暗号プリミティブが疑似ランダム関数 (疑似ランダム置換) であることを利用するのではなく、その写像の一様性のみを利用する方式も存在する。Daemen らは、メッセージ処理を行う関数として、AES のラウンド関数 4 段 (鍵無し) を用いる Pelican を提案した [10, 11]。Pelican 2.0 [11] の安全性は証明されていない。しかし、現実的な攻撃も報告されていない。Minematsu らは、同じく AES のラウンド関数 4 段 (鍵付き) を用いる PC-MAC-AES を提案した [22]。PC-MAC-AES は、ベースとなる関数の最大差分確率を前提として安全性が証明されている。したがって、安全性の

観点では、Pelican よりも PC-MAC-AES が優れている。いずれのアルゴリズムも、AES 以外の軽量ブロック暗号をベースに構成することが可能であるが、事前に最大差分確率の評価が必須である。

実装性能では、Pelican や PC-MAC-AES は、いずれも漸近的な性能が CMAC-AES の 2.5 倍である。ただし、いずれも事前処理や最終処理に AES の暗号化 1 回以上の処理を行うため、メッセージ長が短い場合にはアドバンテージが小さくなる。また、PC-MAC-AES の処理速度は拡大鍵の量とトレードオフの関係にあり、漸近的な処理速度に近づくためには、メモリ使用量が増大する。したがって、実装性能の観点では Pelican が優位である場合が多い。

これらの他に、独自の暗号プリミティブを用いる方式として、Mouha らは非線形置換を用いる Chaskey を提案した [23]。Chaskey は 1-key Even-Mansour ブロック暗号の CMAC と解釈することが可能である。また、非線形置換は Skein, SipHash [1] と同様、ARX 演算をベースにしている。事前処理、最終処理が無いため、短いメッセージに対して効率的であると考えられる。表 2.21 は [23] で報告されている、Chaskey の処理速度である。

表 2.21　Cortex-M 上での Chaskey の処理速度 (cycles/Byte) [23]

	データ長	
	16	128
Cortex-M0	21.3	18.3
Cortex-M3/M4	10.6	7.0

参考文献

[1] Jean-Philippe Aumasson and Daniel J. Bernstein. “SipHash: A Fast Short-Input PRF”. *INDOCRYPT*, LNCS 7668, pages 489–508, Springer, 2012.

[2] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. “Keying Hash Functions for Message Authentication”. *Advances in Cryptology, CRYPTO'96*, LNCS 1109, pages 1–15, Springer, 1996.

[3] Daniel J. Bernstein. “The Poly1305-AES Message-Authentication Code”. *Fast Software Encryption, FSE'05*, LNCS 3557, pages 32–49, Springer, 2005.

[4] Daniel J. Bernstern. “Poly1305-AES speed tables”. `http://cr.yp.to/mac/speed.html`.

[5] Gyido Bertoni, Joan Daemen, Michaël Peeters and Gilles Van Assche, “Sponge functions,” ECRYPT Hash Workshop, May 2007.

[6] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche, “On the security of the keyed sponge construction”. *Symmetric Key Encryption Workshop, SKEW'11*, 2011.

[7] John Black, Shai Halevi, Hugo Krawczyk, Ted Krovetz, and Phillip Rogaway. “UMAC: Fast and Secure Message Authentication”. *Advances in Cryptology, CRYPTO'99*, LNCS 1666, pages 216–233. Springer, 1999.

[8] M. Boesgaard, T.Christensen and E. Zenner, “Badger – A fast and provably secure MAC.” *Applied Cryptgraphy and Network Security*, LNCS 3531, pagets 176–191, Springer, 2005.

[9] J. Lawrence Carter and Mark N. Wegman. “Universal Classes of Hash Functions”. *J. Comput. Syst. Sci.*, 18(2):143–154, 1979.

[10] Joan Daemen and Vincent Rijmen. “A New MAC Construction ALRED and a Specific Instance ALPHA-MAC”. *Fast Software Encryption, FSE'05*, LNCS 3557, pages 1–17, Springer, 2005.

[11] Joan Daemen and Vincent Rijmen. “The MAC Function Pelican 2.0”. *IACR Cryptology ePrint Archive*, 2005:88, 2005. `https://eprint.iacr.org/2005/088.pdf`.

[12] Morris Dworkin. “Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication”. NIST special publication 800-38b, May 2005. `http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf`.

[13] Morris Dworkin. “Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC”. NIST special publication 800-38d, November 2007. `http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf`.

[14] Shimon Even and Yishay Mansour. “A Construction of a Cioher From a Single Pseudorandom Permutation”. *Advances in Cryptology, ASIACRYPT*, LNCS 739, pages 210–224. Springer, 1991.

[15] Shay Gueron. “AES-GCM software performance on the current high end CPUs as a performance baseline for CAESAR competition.” *Directions in Authenticated Ciphers, DIAC 2013*. 2013.

[16] Niels Ferguson. "Authentication weaknesses in GCM". Comments submitted to NIST Modes of Operation Process, May 2005.

[17] Shai Halevi and Hugo Krawczyk, "MMH: Software message authentication in the Gbit/second rate". *Fast Software Encryption, FSE'97*, LNCS 1267, pages 172–189, Springer, 1997.

[18] Tetsu Iwata and Kaoru Kurosawa. "OMAC: One-Key CBC MAC". *Fast Software Encryption, FSE'03*, LNCS 2887, pages 129–153. Springer, 2003.

[19] Antoine Joux. "Authentication Failures in NIST version of GCM". Comments submitted to NIST Modes of Operation Process, June 2006.

[20] Ted Krovetz. "Message Authentication on 64-Bit Architectures". *Selected Areas in Cryptography*, LNCS 4356, pages 327–341. Springer, 2006.

[21] Ted Krovetz. "UMAC Performance". `http://web.cs.ucdavis.edu/~rogaway/umac/2004/perf04.html`

[22] Kazuhiko Minematsu and Yukiyasu Tsunoo. "Provably Secure MACs From Differentially-uniform Permutations and AES-based Implementations," *Fast Software Encryption, FSE'06*, LNCS 4047, pp. 226–241, 2006.

[23] Nicky Mouha, Bart Mennink, Anthony Van Herrewege, Dai Watanabe, Bart Preneel, and Ingrid Verbauwhede. "Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers." *Selected Areas in Cryptography, SAC'14*, 2014.

[24] Gordon Procter and Carlos Cid. "On Weak Keys and Forgery Attacks Against Polynomial-Based MAC Schemes," *Fast Software Encryption, FSE'13*, LNCS 8424, pp. 287–304, Springer, 2014.

[25] Markku-Juhani O. Saarinen. "Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes." *Fast Software Encryption, FSE'12*, 2012.

[26] James M. Turner. "The Keyed-Hash Message Authentication Code (HMAC)". FIPS PUB 198-1, National Institute of Standards and Technology, July 2008. `http://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf`.

[27] Mark N. Wegman and J. Lawrence Carter. "New Hash Functions and Their Use in Authentication and Set Equality". *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.

[28] ISO/IEC 9797-1:2011, Information technology – Security tecniques – Message authentication codes (MACs) – Part 1: Mechanisms using a block cipher, 2011.

[29] ISO/IEC 9797-2:2011, Information technology – Security tecniques – Message authentication codes (MACs) – Part 2: Mechanisms using a dedicated hash-function, 2011.

[30] ISO/IEC 9797-3:2011, Information technology – Security tecniques – Message authentication codes (MACs) – Part 3: Mechanisms using a universal hash function, 2011.

2.6 認証暗号

2.6.1 認証暗号の安全性

2.6.1.1 はじめに

共通鍵暗号を用いて、送りたい平文の暗号化と、改ざん検知のための認証タグの付与とを同時に行う、認証暗号（Authenticated Encryption, AE）と呼ばれる機能が知られている。本章では、AE について、特に安全性の側面から調査した結果を報告する。

まず、AE の形式的な定義、およびその安全性の概念として知られているものを整理し 2.6.1.2 章で報告する。基本的に AE の安全性は、暗号文が平文の関する情報をどれだけ隠しているか（秘匿性）と、不正な暗号文をどれだけ正確に検知できるか（真正性）の二つの軸で評価されている [7, 46]。ただし、これらを複合した単一評価軸の存在や、平文以外の入力変数である初期ベクトルなどの形式、生成モデルの違いによりいくつかバリエーションを生じている。これらの違いを意識した安全性の比較が必要である。

次に 2.6.1.6 章以降にて具体的な構成方法を紹介する。AE の構成方法は多岐にわたり、特に用いる暗号学的プリミティブによって全体構成が大きく変化する。ここではもっとも普及しているアプローチの一つである、ブロック暗号をベースとした方式、すなわちブロック暗号利用モード（以下モード）により実現される方式に絞って説明を行い、これらが満たす安全性を示す。

2.6.1.2 認証暗号の形式と安全性

■基本的な入出力　まず、認証暗号の入出力について解説する。認証暗号の処理は一般に暗号化と復号からなる。秘密鍵 K を共有する 2 者間において、暗号化関数の入力は、もっとも典型的な場合、

- 初期ベクトル（Initial Vector, IV）N
- 平文 M
- ヘッダ H

となる。ここで、初期ベクトル N は暗号化のために補助的に用いる変数であり、通常暗号文と共に通信される（従って受信側は初期ベクトルを同期する必要がない)。初期ベクトルの長さは固定の場合も可変長の場合もある。典型的な生成方法は乱数によるものか、暗号化側が保持し、逐次更新する状態変数（カウンターなど）を用いるもの、あるいはその両方の組み合わせによるものである。

平文 M は暗号化の対象となる情報であり、一般に可変長の系列である。

ヘッダ H は associated data (AD) とも呼ばれ、暗号化はされないものの改ざんは防ぎたい情報のことを指す。例えば通信プロトコルのバージョン、パラメータ、中継ポイントでのルーティング情報などがある。こちらも一般に可変長の系列である。

なお厳密にはヘッダの存在しない方式を AE と呼び、ヘッダがある方式を AEAD（AE with AD）と呼ぶことがあるが、本稿では区別せず AE と呼ぶ。AEAD は方式によってはヘッダが存在せず、長さ 0 の変数と解釈して処理を行うことが可能であり、その意味では AE を包含する概念といえる。さらに、平文 M が存在しない場合を認める方式もあり、この場合の意図する処理はヘッダ H に対する、IV 付きのメッセージ認証コード（Message Authentication Code, MAC）となる。

暗号化処理の出力は、

- 暗号文 C
- タグ T

となる。暗号文 C の長さは通常 M と同じであり、タグ T は固定長である。送信する情報は (N, A, C, T) の 4 つ組となる。

復号処理の入力は上記 4 つ組であり、出力結果は、もし送信された情報が改ざんされていないと判断（受理）された場合には復号された平文 M となり、改ざんがあったと判断した場合は、単一のエラーメッセージとなる。

■入出力形式のバリエーション　基本的な AE には IV は必須であるが、方式によってはこれを不要とするものがある。例えば ANSI のスマートメータ関連規格（C12.22）において定義されている EAX-prime という方式では、IV とヘッダを組み合わせた変数を Cleartext と呼んでいる。また、いわゆる Deterministic AE (DAE), On-line AE (OAE) と呼ばれる AE のクラスにおいては、IV は存在せず、もし存在する場合には暗黙にヘッダに含まれるものとされていることが多い。

2.6.1.3　安全性の概念 – IV 付きの場合

上述のように、安全性の概念は典型的に秘匿性（Privacy）と完全性（Authenticity・Integrity）に分けて説明される。秘匿性とは、送信内容である (N, A, C, T) を得た攻撃者が元の平文 M に対する情報を得ることの困難性を表す指標であり、より端的には、暗号化関数の出力である (C, T) と同じ長さの乱数との判別困難性をもって表される。完全性とは、攻撃者が改ざんに成功することの困難性を表す指標である。ここで、改ざんとは、観測した正規の (N, A, C, T) とは異なる $(N', A', C', T') \neq (N, A, C, T)$ を、鍵を知ることなく生成し、これを受信者が受理する事象を指す。完全性は改ざん成功確率を攻撃者のクラスに関して最大値をとることで評価される。

よりフォーマルに記載するために、以下の表記を導入する。まず、$\mathcal{A}^{O_1, O_2, \ldots, O_c}$ を攻撃者 $\mathcal{A}$ が c 個のオラクル $O_1, \ldots, O_c$ に任意の順序でアクセスする環境を示すものとする。次に $\mathsf{AE}[\tau]$ を、τ-bit のタグを持つ AE であるとし、その暗号化と復号の関数をそれぞれ $\mathsf{AE}\text{-}\mathcal{E}_\tau$ と $\mathsf{AE}\text{-}\mathcal{D}_\tau$ とする。秘匿性の定義は以下で与えられる。まず $\mathsf{AE}[\tau]$ への nonce-respecting な q 選択平文攻撃とは $\mathsf{AE}\text{-}\mathcal{E}_\tau$ に対して $(N_1, H_1, M_1), \ldots, (N_q, H_q, M_q)$ を逐次的・適応的に与えて、$(C_1, T_1), \ldots, (C_q, T_q)$ を得ることをいう。ただしどの $i < j$ についても $N_i \neq N_j$ となることが条件である。ここで \$ を、(N, H, M) が与えられたもとで常に (C, T) と同じ長さの乱数を返す、ランダムビットオラクルであるとする。すると AE へ nonce-respecting な選択平文を行う攻撃者 $\mathcal{A}$ に対する PRIV アドバンテージは

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1].$$

と定義される。

次に完全性を定義する。$\mathcal{A}$ が $\mathsf{AE}[\tau]$ に対する選択暗号文攻撃を行う場合、$\mathsf{AE}\text{-}\mathcal{E}_\tau$ と $\mathsf{AE}\text{-}\mathcal{D}_\tau$ の両方に任意の順序でアクセスできる。$\mathcal{A}$ は nonce-respecting な選択平文クエリを $\mathsf{AE}\text{-}\mathcal{E}_\tau$ へ行うが、$\mathsf{AE}\text{-}\mathcal{D}_\tau$ には IV に関する制約はない。つまり暗号化クエリで用いた IV を復号クエリに用いてもよいし、復号クエリで重複した IV を用いてもよい。ただし自明な答えが返ってくる、暗号化で聞いた結果をそのまま復号に与えることだけは禁じる。このような攻撃者 $\mathcal{A}$ について、AE の完全性は、

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau, \mathsf{AE}\text{-}\mathcal{D}_\tau} \text{ forges }]$$

で定義される。ここで右辺は、エラーシンボルである $\perp$ 以外を $\mathsf{AE}\text{-}\mathcal{D}_\tau$ から得るのイベントの確率を指す。

なお PRIV/AUTH といった表記については論文によっては異なる名称をとる場合もあるので注意が必要である。後述の IV なしのケースについても同様。

2.6.1.4 安全性の概念 – IV なしの場合

■Deterministic AE　IV が存在しない場合、暗号化の入力が (H, M)（もし H が存在すれば）で、出力が (C, T)、送信内容が (H, C, T) となる。また復号処理は (H, C, T) を入力とし、受理すれば M を出力、そうでなければ $\perp$ 出力となる。

このような AE の安全性については、大きく二つのバリエーションがある。一つ目は、Privacy については、平文の一致情報以上は漏らさないことを求める方式である。Authenticity については (H, C, T) に対する改ざん困難性を要求する。この概念は最初に Rogaway と Shrimpton によって提案され、Deterministic AE (DAE) と呼ばれることから、DAE security とも呼ばれている。

IV 付きの場合と同様に PRIV/AUTH で評価する場合について述べる。まず秘匿性は、$\mathsf{AE}[\tau]$ を DAE とみなし、AE-$\mathcal{E}_\tau$ へクエリ (H, M) を重複して行わない $\mathcal{A}$ について,

$$\mathtt{Adv}^{\mathtt{dpriv}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1]$$

で評価する。一方完全性は IV 付きのケースと同様

$$\mathtt{Adv}^{\mathtt{dauth}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau, \mathsf{AE}\text{-}\mathcal{D}_\tau} \text{ forges }],$$

で評価する。DAUTH の forge の意味は、non-trivial な復号のクエリ (H, C, T)（すなわち (H, M) を暗号化クエリして (C, T) を得ていない）について $\perp$ 以外のレスポンスを得ることを指す。それぞれの指標を DPRIV, DAUTH とここでは呼ぶことにする。なお Rogaway と Shrimpton は同時にこの二つをまとめた指標として DAE-advantage を提案している。これは、

$$\mathtt{Adv}^{\mathtt{dae}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau, \mathsf{AE}\text{-}\mathcal{D}_\tau} \Rightarrow 1] - \Pr[\mathcal{A}^{\$, \perp} \Rightarrow 1]$$

という指標である。これは、攻撃者が DAE の暗号化と復号に両方アクセスできるもとで、自明な質問をしないで、random-bit oracle と $\perp$-oracle (常に $\perp$ を返すオラクル）の組と、実際の DAE 暗号化、復号の組との判別を行う困難性を示すものである。両者は基本的に等価な関係にあり、DPRIV と DAUTH（の上界）が求まれば、DAE-advantage の上界が求まり、またその逆も可能であることが示されている [48]。単一指標のほうがシンプルな表現ではあるが、従来指標との整合性、および実際の証明手続きを考えると、二軸での指標にも実用的価値が見いだせると思われる。

DPRIV が求めるものは、本質的に暗号文のどのビットも平文全体の情報を反映することであり、従って DAE には原理上平文全体を読み込まない限り暗号文の最初のブロックが計算できず、従ってオンライン処理（1 パス処理）が不可能である。

■On-line AE　もう一つのケースが、秘匿性において異なる平文間の prefix の一致だけ漏れることを許容し、それより後ろは漏らさない、とするものである。このような機能は一般的に On-line Cipher と呼ばれ、Bellare らの研究 [5] に端を発するものである。認証暗号として完全性も満たすよう拡張された方式も提案されており、On-line AE (OAE) と呼ばれている。

まず秘匿性は、$\mathsf{AE}[\tau]$ を OAE とみなし、AE-$\mathcal{E}_\tau$ へクエリ (H, M) を重複して行わない $\mathcal{A}$ について,

$$\mathtt{Adv}^{\mathtt{opriv}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau} \Rightarrow 1] - \Pr[\mathcal{A}^{\$^O} \Rightarrow 1]$$

で評価する。一方完全性は IV 付きのケースと同様

$$\mathtt{Adv}^{\mathtt{oauth}}_{\mathsf{AE}[\tau]}(\mathcal{A}) \stackrel{\text{def}}{=} \Pr[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{AE}\text{-}\mathcal{E}_\tau, \mathsf{AE}\text{-}\mathcal{D}_\tau} \text{ forges }],$$

で評価する。本質的に等価な別の定義として、$\mathtt{Adv}^{\mathtt{oauth}}$ はまた、$\perp$ を常にエラーシンボル $\perp$ を返すオラクルと定義したうえで、$(\mathsf{AE}\text{-}\mathcal{E}_\tau, \mathsf{AE}\text{-}\mathcal{D}_\tau)$ と $(\mathsf{AE}\text{-}\mathcal{E}_\tau, \perp)$ との判別のアドバンテージと定義されることもある [11]。

なお $\$^O$ は random-bits on-line oracle と呼ばれる、prefix が一致する部分のみ同じ乱数を返すオラクルである。より正確には、ヘッダが同じ二つの入力 (H, M), (H, M'), $M \neq M'$ について、M, M' を n-bit ブロックに分割した表現を $M = M[1], \ldots, M[m]$, $M' = M'[1], \ldots, M'[m']$ とし、Length of Longest common prefix (LLCP) を、$\mathrm{LLCP}_n(M, M') = \max_i\{M[1], \ldots, M[i] = M'[1], \ldots, M'[i]\}$ とする。対応する暗号文が $C = C[1], \ldots, C[m]$, $C' = C'[1], \ldots, C'[m']$ のとき、$\mathrm{LLCP}_n(M, M') = i$ なら まず $C[1], \ldots, C[i] = C'[1], \ldots, C'[i]$ がランダムに選択され、残りの系列が独立かつランダムに選ばれる。このようなオラクルは過去のクエリを保持しその都度サンプリング (lazy sampling) を行うことで実現可能である。

また、Fleischmann ら [11] は DAE のケースと同様に、OPRIV と OAUTH をまとめた指標である CCA3-security を提案し、OPRIV と OAUTH との等価性を説明している。

DAE とは異なり、OAE は秘匿性の部分は On-line cipher と同等の安全性要件と同じであるため、オンライン処理が可能である。

■Nonce-misuse との関連　DAE, OAE ともに、IV がヘッダの一部に含まれているケースを考えることが可能である。この場合、上記の安全性基準は、IV が nonce として暗号化に用いられている限りは通常の IV 付き AE の安全性を保証し、IV の重複が暗号化で発生する場合には DAE/OAE 本来の安全性が保証される、ということを意味する。この性質は、特に DAE について Rogaway と Shrimpton により Misuse-resistent AE (MRAE) [48] と呼ばれているが、OAE に関しては達成できる安全性が DAE よりも弱いため、OAE も含めて MRAE と呼称すべきかどうかについては議論がある（例えば CAESAR メーリングリスト [1] の議論参照)。

2.6.1.5　計算量的仮定

上記の安全性概念・基準を達成するにあたり、用いられるブロック暗号に対する計算量的仮定としては以下のものがある。ブロック暗号のブロックサイズを n ビット、またその暗号化関数を E_K, 復号関数を D_K とすると、

- 疑似ランダム関数（Pseudorandom Function, PRF)：選択平文攻撃において n-bit ランダム関数との計算量的判別困難性を有する鍵付き関数。
- 疑似ランダム置換（Pseudorandom Permutation, PRP)：選択平文攻撃において n-bit ランダム置換との計算量的判別困難性を有する鍵付き関数。
- 強疑似ランダム置換（Strong Pseudorandom Permutation, SPRP)：選択暗号文攻撃において n-bit ランダム置換との計算量的判別困難性を有する鍵付き関数。
- 関連鍵安全性（Related-key Security)：攻撃者が関連鍵を入力できる環境における、上記の計算量仮定のいずれか。例えば定数 c を鍵差分として入力できる PRP の場合、$K, K' = K \oplus c$ においてペア $(E_K, E_{K'})$ と ペアの独立なランダム置換 $(\mathsf{P}, \mathsf{P}')$ の判別困難性を意味する。

2.6.1.6　方式説明における記法

次節から具体的な方式を取り上げ、それらの概略と、証明可能安全性について述べる。AE の実現方法は多様であるため、ここではブロック暗号をベースとした暗号利用モードにより実現されている例を中心に取り扱う。仕様の解説はおおまかなものにとどめる。また、安全性の評価を簡潔にするため、以下ではすべて n-bit ブロック暗号を用いるものとし、

- q：暗号化クエリの回数
- q_v：復号クエリの回数
- σ_p：暗号化のクエリ (N, A, M) のトータルのブロック長
- σ_a：暗号化のクエリ (N, A, M) および復号のクエリ (N, A, C, T) のトータルのブロック長
- τ：タグのビット長

というパラメータ群を用いて攻撃者 $\mathcal{A}$ を定義し、$\mathcal{A}$ に対する安全性評価指標（バウンド）を表すことにする。攻撃者 $\mathcal{A}$ の計算量を便宜的に t とするが、本稿におけるバウンドの式では陽には現れないため省略する。また特段断らない限り、バウンド中の定数は略すこととする。実際の定数、および具体的なパラメータの設定においては、必要に応じて引用文献を参照のこと。認証暗号方式 XXX について、$\mathtt{XXX}[E, \tau]$ を、用いるブロック暗号が E で、タグ長が τ ビットとした実現例とする。多くの場合 $1 \leq \tau \leq n$ である。また、$\mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}')$, $\mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}')$ を $\mathcal{A}$ から求まる $\mathcal{A}'$ による、E に対する疑似ランダム置換、および強疑似ランダム置換との判別可能性を表すものとする。$\mathcal{A}'$ のパラメータは、計算量を含め $\mathcal{A}$ から決まるため方式ごとに定義が必要だが、以下では、$\mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}')$ 中の $\mathcal{A}'$ はすべて $O(\sigma_p)$（定数は一般に小さい）回の CPA クエリを行う、計算量 $O(t\sigma_p)$ の攻撃者となる。同様に $\mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}')$ 中の $\mathcal{A}'$ はすべて $O(\sigma_a)$ 回の CCA クエリを行う、計算量 $O(t\sigma_a)$ の攻撃者となる。使うブロック暗号の鍵の数が 2 以上の場合、一般的にこれらの項にも係数が出てくるが、こちらも省略するものとする。

2.6.1.7 IV 付き、レート 2 の方式

平文 M の 1 ブロックあたりの処理に必要なブロック暗号の回数をレートと呼ぶことにする。このような方式は、一般的に安全な暗号化のモード（カウンターモードなど）とメッセージ認証コード（CMAC など）を異なる鍵で適切に組み合わせることで構成可能であり、これを generic composition と呼ぶ。以下で説明するものの中には generic composition と類似した構成も含まれるが、鍵が共通であるため、generic composition の安全性結果（[6, 41] など）を直接引用することはできない。

■CCM　設計者: Housley, Whiting, Ferguson により 2002 年に作られた [54]。
構成：CBC-MAC で (N, H, M) を処理して中間タグ T' を生成したのち、N および H, M の長さ情報からカウンターモード暗号化の IV を生成し、M と T' を連結した系列を暗号化し、暗号文 C とタグ T とする。いわゆる MAC-then-Enc という generic composition の形式をとる（ただし鍵は単一である）。このため、本質的に On-line 処理ができない。IV 長は 1 ブロック未満に制限されている。また、CBC-MAC 入力のフォーマットが本来不要な複雑さを持つ、という問題がある。
安全性：Johnson [23] により以下の安全性証明がなされている。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{CCM}[E,\tau]}(\mathcal{A}) \leq \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{CCM}[E,\tau]}(\mathcal{A}) \leq \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

つまり、クエリの総ブロック長が $2^{n/2}$ より十分小さく、復号クエリの回数が 2^τ より十分小さい限り、CCM は PRIV/AUTH の双方の意味において、用いるブロック暗号の疑似ランダム性に帰着される安全性を有するといえる。

このタイプのバウンドは IV 付き AE の中でもっともよく見られるものである。

■GCM　設計者: McGrew と Viega により 2004 年に作られた [30]。
構成：$n = 128$-bit のブロック暗号によるカウンターモード GCTR と、有限体 $\mathrm{GF}(2^n)$ 上の乗算を用いたユニバーサ

ルハッシュ関数である GHASH とを組み合わせている。全体構成としては Enc-then-MAC の構成に近い。IV N は任意長をとれるが、特に $|N| = 96$ の場合、$I = N$ とし I の下 32-bit をインクリメントした値を初期値とした GCTR で M を暗号化し C を得たのち、GHASH を (A, C) へ適用し、$E_K(I)$ との XOR によりタグ T を生成する。これ以外の長さでは $I = \mathrm{GHASH}(N)$ としたのち同様の処理を行う。なお、処理量としては平文 m ブロック、ヘッダ a ブロック、IV x ブロックにつき $m + 1$ 回のブロック暗号コール、$a + m + x$ 回の GF 乗算を必要とする。乗算のコストと実装規模（コードサイズ、事前計算量など）は無視できないため、ブロック暗号のレートとしては 1 であるが、トータルの計算コスト、実装規模は下記のレート 1 の方式と同等ととらえることはできない。

安全性：当初、McGrew, Viega により安全性証明がなされた [31] が、後に岩田らにより誤りが発見され、成功確率は現実的ではないが理論的攻撃が示された [20]。これは 96-bit 以外の IV を用いる時にカウンタ衝突確率の上界評価が当初の証明より大幅に増加することを利用している。同時に、証明の誤りを修正した以下のバウンドが示された。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{GCM}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n} + \frac{2^{22} q \sigma_p \ell_N}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{GCM}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{2^{22} (q + q_v) \sigma_a \ell_N}{2^n} + \frac{q_v \ell_A}{2^\tau}$$

ただし ℓ_N と ℓ_A は暗号化および復号で用いた最大の IV ブロック長と、最大のヘッダブロック長である。上記のバウンドは特別に大きい係数 2^{22} のみ省略せずに記載している。またこの係数は IV を 96-bit に固定することで 1 とすることが可能であるため、安全性を確保する上ではこの設定が望ましい。

■EAX　設計者：Bellare, Rogaway, Wagner により 2004 年に作られた [7]。
構成：CMAC で N を処理した結果 $\widetilde{N}$ を初期値としたカウンターモードで M を暗号化し、C を得たのち、H, C を個別に CMAC で処理した結果の XOR をとり、さらに $\widetilde{N}$ との XOR もとることでタグ T を生成する。N は任意の可変長変数である。CMAC は 3 回コールされるが、それぞれ最初に異なる定数ブロックを挿入することで、独立な疑似ランダム関数として振る舞うようにしている。いわゆる MAC-then-Enc という generic composition の形式をとるが、鍵は単一である。
安全性：Bellare, Rogaway, Wagner により安全性証明がなされている。この証明は AUTH のバウンドが $q_v = 1$ のケースについてのみ扱っており、汎用的な変換方法を用いて $q_v \ge 1$ のケースのバウンドに変換すると、次数 3 の項 $\sigma^2 q_v / 2^n$ が出現するためバースデーバウンドではなくなることが知られていたが、最近峯松らの結果により改善された [36]。ここでは改善されたバウンドで示す。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{EAX}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{EAX}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

■CLOC と SILC　設計者：Iwata, Minematsu, Guo, Morioka による CLOC [17, 16] と、Iwata, Minematsu, Guo, Morioka, Kobayashi による SILC [18, 19] がある。いずれも CFB と CBC-MAC との組み合わせをベースとし、事前計算を要する入力マスクを無くし、実行中に必要なメモリ量を減らす構造をとり、また 64-bit ブロック暗号の利用も定義するなど、ローエンドデバイスでの動作を意識した方式となっている。CLOC は組み込みソフトウェアを、SILC は小規模ハードウェアを主なターゲットとおいている。安全性：CLOC は [17, 16] により、SILC は [19] により、下記のタイプの標準的なバースデーバウンド安全性が示されている。また、AUTH に関しては Nonce が暗号化で再利用されても安全性が保証されるという特徴を持つ。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{CLOC}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{CLOC}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{SILC}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{SILC}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

■CHM と CIP　ここまで示したのはすべて $2^{n/2}$ ブロックの質問によりバウンドが 1 になり安全性保証がなくなる、安全性に関していわゆるバースデー限界のある方式である。一方岩田 [15, 14] により、バースデー限界を超えた安全性を保証する方式が提案されている。CHM と CIP の二つがあり、いずれもカウンターモードの変種である CENC と、代数的なユニバーサルハッシュ関数との組み合わせである。カウンターモードは鍵ストリーム系列と乱数との判別が $O(2^{n/2})$ ブロック出力させることで可能となるのに対し、CENC は周期的にブロック暗号を追加コールし、その結果をカウンターモード出力へ加算することでバースデー限界を超えた安全性を保証するものである。正整数 w を用いて周期を 2^w（2^w ブロックおきに追加のコールを行う）とした場合、CENC のレートは $1 + 1/2^w$ となり、安全性のバウンドはおおよそ $\sigma^3/2^{2n} + \sigma/2^n$ となる。w は大きいほうがレートが下がるが安全性バウンドと事前計算量などに影響を及ぼすため、$n = 128$ のときは $4 \sim 8$ 程度が推奨される。CHM と CIP についてもほぼ同様の安全性バウンドが得られる。暗号化のレートもほぼ CENC 同様だが、平文ブロック数の GF(2^n) 乗算を要するため、前述のルールに従うとレートは $2 + 1/2^w$ となる。

2.6.1.8　IV 付き、レート 2 未満の方式

■OCB　設計者：正確には 3 つのバージョンが知られており、OCB1,2,3 と呼称される。OCB1 は Rogaway [47] により 2001 年に, OCB2 は同じく Rogaway [45] により 2004 年に、OCB3 は Krovetz と Rogaway [26] により 2011 年に作られた。

構成：ECB 暗号化の上下のブロックをマスク系列で XOR している。マスク系列は、IV N と何番目のブロックかを表すインデックス $i = 1, 2, \ldots$ とをブロック暗号で処理して、i についてシーケンシャルに生成する。平文 M をマスク付き ECB 暗号化した出力が暗号文 C となり、タグ T は平文の全ブロックの XOR（チェックサムと呼ばれる）を特別なマスクを入力側に付けた 1 ブロック ECB で暗号化することで得られる。これはヘッダが存在しないときの処理であり、ヘッダがある場合、並列実行可能な MAC である PMAC をヘッダに適用した結果と上記の T との XOR をタグとする。PMAC は上記のマスク付き ECB の出力全ブロックの XOR をもう一度マスク付き 1 ブロック ECB 暗号化するものである。復号においてはマスク付き ECB の復号をしたのち、得られた平文のチェックサムを暗号化して、タグとの一致をチェックする。この処理にはブロック暗号の復号関数を要する。この構造により、レート 1 を達成している。

OCB の各バージョンでマスク系列生成方式に違いがある。OCB1,3 は Gray code をベースとしており、基準となる n 個のブロック値をブロック暗号を用いて事前計算し、Gray code が示す順序に従って基準のブロック値を逐次的に XOR していくことでマスクを生成するのに対し、OCB2 はほぼ事前計算なしに逐次的に GF(2^n) 上の 2 倍算を繰り返すことでマスクを生成する。

安全性：各提案論文 [47, 45, 26] によりそれぞれのバージョンの安全性証明がなされている。基本的にはいずれも以下

の形で示される。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{OCB}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{OCB}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

ただし、調査した限りでは、OCB1 のみ上記の AUTH バウンドにおいて $q_v = 1$ のケースしか示されていないようである。

なお、最近の結果として、青木と安田 [3] により、OCB の各バージョンとも強疑似ランダム置換よりも弱い条件に帰着できることが示された。上記のバウンドにおいて PRIV は PRP のみで証明できることはほぼ自明であるが、[3] によると、AUTH においてもブロック暗号復号の結果に対する予測不能性（系列全体がランダムでなくとも満たしうる性質）があればよいことが分かる。

■類似方式　ほぼ OCB1 と同時期に発表された方式として Julta [25] による IAPM, IACBC, Gligor, Donescu [12] による XCBC がある。これらは構造的には OCB と同じ（もしくは ECB の内部にさらにブロック間のチェインを挟むものもある）であるが、マスク生成の部分に関して OCB がもっとも洗練されているといえる。

OCB と類似した構造で、マスクを用いずに ECB のブロックをチェインさせて、すなわち CBC 暗号化のような処理を行ってレート 1 の AE を達成しようとする試みもある（例えば PCBC とその変種 [33, 38], IOBC [39],IOC [44]）。OCB 以前に考えられた方法が多い。また [39] によるとそのほぼすべてに攻撃が発見されており、現在のところ安全性証明が与えられた方式はないとみられる。

■CCFB　設計者: Lucks [28] により 2005 年に作られた。
構成：ブロック暗号の入出力の一部のみを用いた CFB モードにより暗号化を行う。CFB で使われない入力部分は処理ブロックのインデックスが与えられ、出力部分は逐次的に XOR をとることでチェックサムとしている。ヘッダが存在しない基本的なバージョンでは、CFB のチェイン値の初期値は IV である。ヘッダが存在するバージョンを CCFB+H と呼ぶが、このバージョンでは、ヘッダを（0^n プリペンドした）CMAC へ適用した結果と IV の XOR をチェインの初期値とする。IV は 1 ブロックの値である。暗号化が終わった時点のチェイン値を暗号化し、チェックサムとの XOR を行いタグとする。タグの長さを τ ビットとすると、チェックサムの長さもこれと等しい。またチェイン値を a-bit とすると $a + \tau = n$ を満たすこととなる。例えば $n = 128$ のケースで $a = 96$, $\tau = 32$ とすることが提案されている。センサーネット系のメッセージ認証コードは 32-bit タグのケースが多く、そのようなケースにフィットすると考えられる。上記の構造により、ブロック暗号 1 回につき a-bit 平文を処理可能であるため、レートは n/a となる。例えば $a = (2/3)n$, $\tau = (1/3)n$ とするとレートは 1.5 となる。原理上は $n/(n-1)$ まで 1 に近づけられるが、タグの短さは AUTH バウンドの劣化に直結するため、適切なバランスを取る必要がある。並列処理が不可能であるが、1 パス暗号化が可能であり、逐次的な処理には適している[*2]。また OCB と異なり、ブロック暗号の暗号化関数のみを用いる。

安全性：Lucks [28] により以下の安全性証明がなされている。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{CCFB}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^a}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{CCFB}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^a} + \frac{1}{2^\tau}$$

ただし AUTH は $q_v = 1$ のケースを扱っている。

[*2] ただし論文のタイトルには Two-pass とある。

■復号関数を用いない方式　OCB がブロック暗号の復号関数を用いるのに対し、レート 1 を保持したままブロック暗号の暗号化関数のみで全体を構成しようとする試みがある。Liting らの iFeed [55, 1] は CBC 暗号化に似た形式（より具体的には暗号化が CBC 復号に類似）を持ち, レート 1 であるが復号が並列処理できない。峯松の OTR [35, 1] では 2 ラウンドフェイステル置換の構造を取り入れることで、2 ブロック単位での並列化が暗号化と復号で可能となっている。いずれも下記に示す標準的なバースデーバウンドの安全性を有している。ブロック暗号の強擬似ランダム性は必要とせず、擬似ランダム性のみを必要とする点が OCB とは異なる。iFeed の安全性証明は Liting ら [56] に記載されている。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{iFeed}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{iFeed}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

また OTR に関しては [35, 34] に記載されている。

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathrm{OTR}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{auth}}_{\mathrm{OTR}[E,\tau]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n} + \frac{q_v}{2^\tau}$$

2.6.1.9　On-line AE 方式

■McOE　設計者: Fleischmann, Forler, Lucks, Wenzel [11] により 2012 年に作られた。
構成：Bellare らによる On-line cipher [4, 5] のアイディアをベースとした構成である。Rogaway と Zhang による、Tweakable ブロック暗号 [27] を用いた On-line cipher の構成方法 TC3 [49] に、さらにメッセージ認証の機能を追加した構成ともとらえることができる。

McOE ではまず、n-bit ブロック暗号 E_K をベースに、n-bit tweak, n-bit ブロックを持つ Tweakable ブロック暗号 $\widetilde{E}_K$ を構成する。構成方法は二つあり、それぞれ McOE-G, McOE-X と呼ばれる。McOE-X で用いる $\widetilde{E}_K$ では、tweak と鍵の XOR により tweak を処理する。具体的には E_K の鍵長 $|K| = n$ であり、平文 M, Tweak T について暗号文は $C = \widetilde{E}_K(T, M) = E_{K \oplus T}(M)$ となる。McOE-G で用いる $\widetilde{E}_K$ では、GF(2^n) 上の要素 X と Y の乗算 $H_Y(X)$ を用いる。具体的には、$|K| = 2n$ であり、$K = (K_1, K_2)$, $|K_i| = n$ と分けたのち、平文 M, Tweak T について暗号文は $C = \widetilde{E}_K(T, M) = E_{K_1}(M \oplus H_{K_2}(T)) \oplus H_{K_2}(T)$ となる。このようにして構成された $\widetilde{E}_K$ を用いて、TC3 の暗号化である Tweak chaining を行う。これは i 番目の平文ブロック $M[i]$ について暗号文ブロック $C[i]$ を $\widetilde{E}_K(S[i], M[i])$ とするものである。$S[i]$ はチェインさせる tweak であり、$S[i+1] = M[i] \oplus C[i]$ として更新する。初期値 $S[0]$ は 0^n である。タグの生成には、最初にヘッダを Tweak chaining で暗号化した結果（の最終ブロック）を Z とし、平文の後ろに Z を連結したのち Tweak chaining で暗号化した結果得られる最終ブロックをタグ T とする。なお平文がブロックサイズの等倍におさまらない場合は、tag-splitting と呼ばれる処理をさらに導入する必要がある（CBC 暗号化における Ciphertext stealing と呼ばれる処理に近い）。タグの長さは常に n bit である。

安全性：[11] により安全性証明がなされている。ここでは簡単のため tag-splitting の不要な、平文が常にブロックサイズの等倍であるケースのバウンドを示す（実際には CCA3 という $\mathtt{Adv}^{\mathtt{opriv}}$ と $\mathtt{Adv}^{\mathtt{oauth}}$ を組み合わせた評価で示して

いるが、証明の内部にて下記のように分解がなされている)。McOE-X と McOE-G それぞれ、

$$\mathtt{Adv}^{\mathtt{opriv}}_{\text{McOE-G}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n},$$
$$\mathtt{Adv}^{\mathtt{oauth}}_{\text{McOE-G}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n},$$
$$\mathtt{Adv}^{\mathtt{opriv}}_{\text{McOE-X}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{rk\text{-}sprp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n},$$
$$\mathtt{Adv}^{\mathtt{oauth}}_{\text{McOE-X}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{rk\text{-}sprp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$

ここで $\mathtt{Adv}^{\mathtt{rk\text{-}sprp}}_{E}(\mathcal{A}')$ は鍵差分 T を入力できるもとでの CCA 攻撃へのアドバンテージ（すなわち、T を tweak とした Tweakable ブロック暗号の CCA セキュリティ）を示す。

McOE-G では安全性がブロック暗号の強疑似ランダム性に帰着されるが、McOE-X では $C = \widetilde{E}_K(T, M) = E_{K \oplus T}(M)$ という暗号化が T ごとに独立と見なせる、という計算量的仮定、すなわち 2.6.1.5 節で述べた関連鍵安全性（Related-key Security）を要する。鍵に Tweak を加算する部分を利用した McOE-G へのアタックが [32] で提案されているが、基本的には計算量 $O(2^{n/2})$ であり、証明自体の決定的な誤りを指摘するものとはなっていない。ただし、このアタックは鍵回復を可能とするものであり、証明が考慮する識別攻撃・改ざん攻撃よりも強い。また [32] は証明における計算量的仮定の置き方に関する問題を示しており、同種の構成を考える際の参考とはなるであろう。

■COPA　設計者: Andreeva ら [2] により 2013 年に作られた。
構成：McOE と異なり、On-line cipher のアイディアを明示的には利用していない。Tweakable ブロック暗号である XEX [45] をベースに、ECB ライクなレイヤーを二つずらして重ねることで構成されている。暗号化のレートは 2 であり、暗号化にはブロック暗号暗号化関数を 2 回、復号にはブロック暗号復号関数を 2 回用いる。CPA-secure な On-line cipher である COPE と、COPE をベースとした On-line AE の COPA が提案されている。タグの長さは n-bit に固定されている。

安全性：[2] で示されている。

$$\mathtt{Adv}^{\mathtt{opriv}}_{\text{COPA}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{oauth}}_{\text{COPA}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$

2.6.1.10　Deterministic AE 方式

■SIV　設計者: Rogaway と Shrimpton [48] により 2006 年に作られた。Deterministic AE(DAE) の最初の提案である。
構成：基本的には平文 M、およびヘッダ A が存在すれば A と M の連結、に対して MAC 関数を適用したのち、得られた出力 V をカウンターモードの初期値として用いて、平文を暗号化する。出力は V とカウンターモードの暗号化結果 C を連結した系列である。復号においては、V を用いて C を復号した後、復号結果 $\widetilde{M}$ を MAC 関数へ適用した結果が V と一致するかでメッセージ認証を行う。MAC 関数とカウンターモードの鍵は独立である。MAC 関数は並列実行可能で、vector-input (pseudorandom) function と呼ばれる形式を持つ、String-to-Vector (S2V) と呼ばれる関数である。これは、一つのバイナリ系列を vector として、vector の系列に対する PRF ととらえることができる[*3]。S2V

[*3] 原理上は Vector-input PRF は容易に単一の可変長入力 PRF と入力の符号化で構成可能であるが、S2V は vector に関するある種のインクリメンタル計算が可能という特徴を持つ。

は可変長（バイナリ系列）入力 PRF を部品として定義される。論文では CMAC を部品としている。レートは 2 であり、ブロック暗号の暗号化関数のみを利用する。タグ長は n bit 以下で設定可能である（が、安全性のバウンドは n の場合のみ扱っているとみられる；$\tau < n$ の場合は $q_v/2^\tau$ がバウンドに加算されると考えられる）。

安全性：[48] により示されている。なお、上述のように [48] では DAE security という単一の指標を中心に説明されているが、DPRIV と DAUTH の二つの指標で導出することが可能である（[48] の Proposition 9 を用いる）。

$$\mathtt{Adv}^{\mathtt{dae}}_{\mathrm{SIV}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{dpriv}}_{\mathrm{SIV}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_p^2}{2^n}$$
$$\mathtt{Adv}^{\mathtt{dauth}}_{\mathrm{SIV}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$

■HBS　設計者: 岩田と安田 [22] により 2009 年に作られた。

構成：SIV がブロック暗号の二つの鍵を用いて構成されているのに対し、HBS ではブロック暗号と Polynomial hashing とを組み合わせ、かつ一つのブロック暗号の鍵のみを用いる。Polynomial hashing の鍵の係数を調節して、ヘッダとメッセージを二つの vector とした vector-input 関数としている。大域的な構成は SIV と似ているが、復号でのタグの検証においてブロック暗号の復号関数を用いるため、全体の安全性はブロック暗号の強疑似ランダム性に帰着される。レートは 1 であり、追加としてヘッダ a ブロック、平文 m ブロックに対して $a+m+2$ 回の $\mathrm{GF}(2^{128})$ 乗算を要する。タグに対してブロック暗号復号関数を適用するため、タグ長は n bit に固定されている。

安全性：[22] により示されている。

$$\mathtt{Adv}^{\mathtt{dae}}_{\mathrm{HBS}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{sprp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$

■BTM　設計者: 岩田と安田 [21] により 2009 年に作られた。

構成：BTM は HBS におけるブロック暗号の復号関数の利用を無くすことを目的に開発された。Polynomial hash の利用などは HBS と同様である。結果として、全体の安全性はブロック暗号の疑似ランダム性に帰着される。レートは 1 であり、追加としてヘッダ a ブロック、平文 m ブロックに対して $a+m-1$ 回の $\mathrm{GF}(2^{128})$ 乗算を要する。タグに対してブロック暗号復号関数を適用しなくてよいため、タグ長は $\tau \le n$ bit に設定することが可能である。

安全性：[22] により示されている（タグ長 n bit のケースであるとみられる）。

$$\mathtt{Adv}^{\mathtt{dae}}_{\mathrm{BTM}[E,n]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{prp}}_{E}(\mathcal{A}') + \frac{\sigma_a^2}{2^n}$$

2.6.1.11　その他

■軽量暗号技術との関連性　ブロック暗号を用いた認証暗号の軽量化に関しては、いくつかのアプローチがある。もっとも全体軽量化に貢献すると思われるのは軽量ブロック暗号を用いることであるが、その多くが 64-bit ブロックサイズであるために、ここで紹介した多くの方式が 32-bit データセキュリティ、すなわち、$2^{32} * 8$ byte $\approx$ 34 Gbyte よりも十分少ないデータ量を処理したところで、鍵の更新が必要となる。例えばカウンターモード（ないしカウンターモードを含んだ AE）を 64-bit ブロック暗号で運用した場合に、PRIV アドバンテージが $\sigma^2/2^{64}$ であるとして、これを 2^{-20} 以下におさえるにはおおよそ $2^{5.5}$ Mbyte のデータ処理の後に鍵更新が必要となる。これは帯域の制限されたセンサーネットなどでセッション鍵生成を頻繁に行う環境であれば実用的だが、一般的にはきわめて制約が強いと思われ

る。なお 128-bit ブロック暗号であれば PRIV を 2^{-20} におさえるのに鍵更新が必要となるデータ処理は $2^{28.5}$ GByte となり、一般的に十分と思われる。

一方、n-bit ブロック暗号で $n/2$-bit 以上のデータセキュリティを保証する AE としては岩田 [15, 14] の方式が知られるのみであり、またこれらの方式は比較的ブロック暗号の外側の処理としてオーバーヘッドが比較的大きい（例えば汎用の $GF(2^n)$ 乗算を有する点で）ため、軽量ブロック暗号のメリットを消してしまう懸念がある。

AE としての複雑さや処理のオーバーヘッドを下げる試みとしては前述の EAX-prime やその改良 [37] があげられる。これらは EAX と比べて、処理の前に必要なブロック暗号のコール回数や、処理中に保持すべきメモリ量を減らしている。またこれらの設計思想をさらに推し進めた CLOC, SILC もある。また、CCFB も安全性のバウンドに強い制約はあるものの、モードとしての処理のオーバーヘッドはかなり小さく、センサーネットでの実装に適することが知られている [24]。ただし、プラットフォームによっては用いるブロック暗号自体の影響が大きく、モードの選択は全体性能において大きな違いをもたらさない可能性もある。

また、一般にセンサーネットで重要とされる消費電力については、計算よりも通信部分の電力消費が大きい。Struik [52] により指摘されているように、組み込み環境で AE による保護を考えるときには、AE 適用による通信量の増分（IV とタグ）を考慮し、ここを小さくするように無駄のないプロトコルを設計することが重要であろう。この場合、IV なし、タグ無しなどの暗号化方式を適切なリスク分析のもと用いることも一つの手段である。

■想定する安全性モデルから逸脱した場合の影響　暗号化、およびメッセージ認証について、安全性のバウンドを超えたデータ量を処理した場合にどのような攻撃が起こりうるかはいくつかの論文で議論されている。例えば McGrew [29] は CTR, CFB, CBC の三つの基本的な暗号化のモードにおいて処理量がバースデーバウンドを超えた場合に、ほぼデータ量の対数に比例して線形に平文ビットが漏れることを示した。また、MAC の場合については Black と Cochran [8] が、一度偽造が成功した場合にその情報をもとにどのような偽造が可能となるかを様々な MAC について調査した。ここでの結果は、該当する MAC を用いた AE についても当てはまることが予想される。ただし AE についてこのような観点から網羅的に安全性評価を試みた研究は見つかっていない。このように安全性の保証を超えた使い方をした場合、いわゆるミスユースに対する安全性の議論は今後重要になるかもしれない。

バースデーバウンドとはやや異なるが、いわゆる弱鍵を利用した攻撃もいくつか提案されている。特に多項式ハッシュ（およびそれを用いている GCM）について数多く報告があり、Sarrinen による cycling attack [51]、これを拡張した Procter と Cid [43] などの研究がある。鍵空間の部分集合 D について、D に鍵が入っているかを $|D|$ よりも少なくテストできるとき、D が弱鍵集合であるというのが従来の定義 [13] であったが、Procter と Cid はこの定義に従った場合、多項式ハッシュの鍵のほぼありとあらゆる部分集合が弱鍵集合とされてしまうことを示した。多項式ハッシュの脆弱性を指摘しているとも受け取れるが、証明可能安全性と矛盾するものではなく、ある意味では弱鍵集合の定義自体の意味を見直す必要があることも示唆している。

2.6.1.12　まとめ

認証暗号の安全性定義と、ブロック暗号に基づく具体的な方式とを調査した結果を報告した。5 章で述べたように、軽量な認証暗号を実現するために部品として軽量ブロック暗号を用いるだけでは解決できない課題がいくつかあり、またそれらの解決には認証暗号より上位のレイヤーでの解決が求められるケースもありそうである。また近年、ブロック暗号を用いず、ハッシュ関数やその部品をベースとする方式や、ブロック暗号のラウンド関数を部品として用いる方式などが提案されてきており、これらの動向にも注意が必要と思われる。

参考文献

[1] CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. `http://competitions.cr.yp.to/caesar.html`.

[2] Elena Andreeva, Andrey Bogdanov, Atul Luykx, Bart Mennink, Elmar Tischhauser, and Kan Yasuda. Parallelizable and authenticated online ciphers. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013*, volume 8269 of *Lecture Notes in Computer Science*, pages 424–443. Springer, 2013.

[3] Kazumaro Aoki and Kan Yasuda. The Security of the OCB Mode of Operation without the SPRP Assumption. In Susilo and Reyhanitabar [53], pages 202–220.

[4] Mihir Bellare, Alexandra Boldyreva, Lars R. Knudsen, and Chanathip Namprempre. Online Ciphers and the Hash-CBC Construction. In Joe Kilian, editor, *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 292–309. Springer, 2001.

[5] Mihir Bellare, Alexandra Boldyreva, Lars R. Knudsen, and Chanathip Namprempre. On-line ciphers and the hash-cbc constructions. *J. Cryptology*, 25(4):640–679, 2012.

[6] Mihir Bellare and Chanathip Namprempre. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. In Tatsuaki Okamoto, editor, *ASIACRYPT*, volume 1976 of *Lecture Notes in Computer Science*, pages 531–545. Springer, 2000.

[7] Mihir Bellare, Phillip Rogaway, and David Wagner. The EAX Mode of Operation. In Roy and Meier [50], pages 389–407.

[8] John Black and Martin Cochran. MAC Reforgeability. In Dunkelman [10], pages 345–362.

[9] Anne Canteaut, editor. *Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers*, volume 7549 of *Lecture Notes in Computer Science*. Springer, 2012.

[10] Orr Dunkelman, editor. *Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers*, volume 5665 of *Lecture Notes in Computer Science*. Springer, 2009.

[11] Ewan Fleischmann, Christian Forler, and Stefan Lucks. McOE: A Family of Almost Foolproof On-Line Authenticated Encryption Schemes. In Canteaut [9], pages 196–215.

[12] Virgil D. Gligor and Pompiliu Donescu. Fast Encryption and Authentication: XCBC Encryption and XECB Authentication Modes. In Mitsuru Matsui, editor, *FSE*, volume 2355 of *Lecture Notes in Computer Science*, pages 92–108. Springer, 2001.

[13] Helena Handschuh and Bart Preneel. Key-Recovery Attacks on Universal Hash Function Based MAC Algo-

rithms. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 144–161. Springer, 2008.

[14] Tetsu Iwata. New Blockcipher Modes of Operation with Beyond the Birthday Bound Security. In Matthew J. B. Robshaw, editor, *FSE*, volume 4047 of *Lecture Notes in Computer Science*, pages 310–327. Springer, 2006.

[15] Tetsu Iwata. Authenticated Encryption Mode for Beyond the Birthday Bound Security. In Serge Vaudenay, editor, *AFRICACRYPT*, volume 5023 of *Lecture Notes in Computer Science*, pages 125–142. Springer, 2008.

[16] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC: Compact Low-Overhead CFB. `http://competitions.cr.yp.to/round1/clocv1.pdf`.

[17] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC: authenticated encryption for short input. *Proceedings of Fast Software Encryption 2014*, 2014:157, 2014.

[18] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, and Eita Kobayashi. SILC: SImple Lightweight CFB. `http://competitions.cr.yp.to/round1/silcv1.pdf`.

[19] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, and Eita Kobayashi. SILC: SImple Lightweight CFB. DIAC: Directions in Authenticated Ciphers, 2014. `http://2014.diac.cr.yp.to/`.

[20] Tetsu Iwata, Keisuke Ohashi, and Kazuhiko Minematsu. Breaking and Repairing GCM Security Proofs. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO*, volume 7417 of *Lecture Notes in Computer Science*, pages 31–49. Springer, 2012.

[21] Tetsu Iwata and Kan Yasuda. BTM: A Single-Key, Inverse-Cipher-Free Mode for Deterministic Authenticated Encryption. In Michael J. Jacobson Jr., Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *Selected Areas in Cryptography*, volume 5867 of *Lecture Notes in Computer Science*, pages 313–330. Springer, 2009.

[22] Tetsu Iwata and Kan Yasuda. HBS: A Single-Key Mode of Operation for Deterministic Authenticated Encryption. In Dunkelman [10], pages 394–415.

[23] Jakob Jonsson. On the Security of CTR + CBC-MAC. In Kaisa Nyberg and Howard M. Heys, editors, *Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 76–93. Springer, 2002.

[24] Marcos A. Simplício Jr., Bruno Trevizan de Oliveira, Paulo S. L. M. Barreto, Cintia B. Margi, Tereza Cristina M. B. Carvalho, and Mats Näslund. Comparison of Authenticated-Encryption schemes in Wireless Sensor Networks. In Chun Tung Chou, Tom Pfeifer, and Anura P. Jayasumana, editors, *IEEE 36th Conference on Local Computer Networks, LCN 2011, Bonn, Germany, October 4-7, 2011*, pages 450–457. IEEE, 2011.

[25] Charanjit S. Jutla. Encryption Modes with Almost Free Message Integrity. In Birgit Pfitzmann, editor, *EUROCRYPT*, volume 2045 of *Lecture Notes in Computer Science*, pages 529–544. Springer, 2001.

[26] Ted Krovetz and Phillip Rogaway. The Software Performance of Authenticated-Encryption Modes. In Antoine Joux, editor, *FSE*, volume 6733 of *Lecture Notes in Computer Science*, pages 306–327. Springer, 2011.

[27] Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable block ciphers. *J. Cryptology*, 24(3):588–613, 2011.

[28] Stefan Lucks. Two-Pass Authenticated Encryption Faster Than Generic Composition. In Henri Gilbert and Helena Handschuh, editors, *FSE*, volume 3557 of *Lecture Notes in Computer Science*, pages 284–298. Springer, 2005.

[29] David A. McGrew. Impossible plaintext cryptanalysis and probable-plaintext collision attacks of 64-bit block cipher modes. *Pre-proceedings of Fast Software Encryption 2013*. Available from `http://eprint.iacr.org/2012/623`.

[30] David A. McGrew and John Viega. The Galois/Counter mode of operation (GCM). NIST Submission, 2004. Available from `http://csrc.nist.gov/groups/ST/toolkit/BCM/modes_development.html`.

[31] David A. McGrew and John Viega. The Security and Performance of the Galois/Counter Mode (GCM) of Operation. In Anne Canteaut and Kapalee Viswanathan, editors, *INDOCRYPT*, volume 3348 of *Lecture Notes in Computer Science*, pages 343–355. Springer, 2004.

[32] Florian Mendel, Bart Mennink, Vincent Rijmen, and Elmar Tischhauser. A Simple Key-Recovery Attack on McOE-X. In Josef Pieprzyk, Ahmad-Reza Sadeghi, and Mark Manulis, editors, *CANS*, volume 7712, pages 23–31. Springer, 2012.

[33] Alfred J. Menezes, Scott A. Vanstone, and Paul C. Van Oorschot. *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 1996.

[34] Kazuhiko Minematsu. Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. *IACR Cryptology ePrint Archive*, 2013:628, 2013.

[35] Kazuhiko Minematsu. Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. In Nguyen and Oswald [42], pages 275–292.

[36] Kazuhiko Minematsu, Stefan Lucks, and Tetsu Iwata. Improved Authenticity Bound of EAX, and Refinements. In Susilo and Reyhanitabar [53], pages 184–201.

[37] Kazuhiko Minematsu, Stefan Lucks, Hiraku Morita, and Tetsu Iwata. Attacks and security proofs of EAX-prime. In Moriai [40], pages 327–347.

[38] Chris J. Mitchell. Cryptanalysis of Two Variants of PCBC Mode When Used for Message Integrity. In Colin Boyd and Juan Manuel González Nieto, editors, *ACISP*, volume 3574 of *Lecture Notes in Computer Science*, pages 560–571. Springer, 2005.

[39] Chris J. Mitchell. Analysing the IOBC Authenticated Encryption Mode. In Colin Boyd and Leonie Simpson, editors, *ACISP*, volume 7959 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2013.

[40] Shiho Moriai, editor. *Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers*, volume 8424 of *Lecture Notes in Computer Science*. Springer, 2014.

[41] Chanathip Namprempre, Phillip Rogaway, and Thomas Shrimpton. Reconsidering Generic Composition. In Nguyen and Oswald [42], pages 257–274.

[42] Phong Q. Nguyen and Elisabeth Oswald, editors. *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, volume 8441 of *Lecture Notes in Computer Science*. Springer, 2014.

[43] Gordon Procter and Carlos Cid. On Weak Keys and Forgery Attacks Against Polynomial-Based MAC Schemes. In Moriai [40], pages 287–304.

[44] Francisco Recacha. Input and Output Chaining. NIST Submission, 2013. Available from `http://csrc.nist.gov/groups/ST/toolkit/BCM/modes_development.html`.

[45] Phillip Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In Pil Joong Lee, editor, *ASIACRYPT*, volume 3329 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2004.

[46] Phillip Rogaway. Nonce-Based Symmetric Encryption. In Roy and Meier [50], pages 348–359.

[47] Phillip Rogaway, Mihir Bellare, and John Black. OCB: A block-cipher mode of operation for efficient authenticated encryption. *ACM Trans. Inf. Syst. Secur.*, 6(3):365–403, 2003.

[48] Phillip Rogaway and Thomas Shrimpton. A Provable-Security Treatment of the Key-Wrap Problem. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 373–390. Springer, 2006.

[49] Phillip Rogaway and Haibin Zhang. Online Ciphers from Tweakable Blockciphers. In Aggelos Kiayias, editor, *CT-RSA*, volume 6558 of *Lecture Notes in Computer Science*, pages 237–249. Springer, 2011.

[50] Bimal K. Roy and Willi Meier, editors. *Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers*, volume 3017 of *Lecture Notes in Computer Science*. Springer, 2004.

[51] Markku-Juhani Olavi Saarinen. Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes. In Canteaut [9], pages 216–225.

[52] Rene Struik. Revisiting design criteria for AEAD ciphers targeting highly constrained networks. DIAC: Directions in Authenticated Ciphers, 2013. `http://2013.diac.cr.yp.to/`.

[53] Willy Susilo and Reza Reyhanitabar, editors. *Provable Security - 7th International Conference, ProvSec 2013, Melaka, Malaysia, October 23-25, 2013. Proceedings*, volume 8209 of *Lecture Notes in Computer Science*. Springer, 2013.

[54] Douglas Whiting, Russ Housley, and Niels Ferguson. Counter with CBC-MAC (CCM). NIST Submission, 2002. Available from `http://csrc.nist.gov/groups/ST/toolkit/BCM/modes_development.html`.

[55] Liting Zhang, Sui Han, Wenling Wu, and Peng Wang. iFeed: the Input-Feed AE Modes. Rump Session of FSE 2013, 2013. slides from `http://fse.2013.rump.cr.yp.to/`.

[56] Liting Zhang, Wenling Wu, Han Sui, and Peng Wang. iFeed[AES] v1. `http://competitions.cr.yp.to/round1/ifeedaesv1.pdf`.

2.6.2　認証暗号の実装性能

本章では、軽量暗号技術の現状調査として、主要な認証暗号の実装性能 (ハードウェア、ソフトウェア) 調査結果をまとめる。

2.6.2.1　調査内容

■Grain-128a　Grain-128 は 2004 年に eSTREAM のハードウェア部門に提案されたアルゴリズムであり、eSTREAM の Winner の一つである。文献 [1] に示されるハードウェア性能を表 2.22 にまとめる。文献 [1] ではゲートカウントの見積もりのみを実施している。

表 2.22　Grain-128a のゲートカウント見積もり

機能	速度モード毎のゲートカウント [gate]					
	1×	2×	4×	8×	16×	32×
暗号化のみ	2145.5	2243	2438	2828	3608	5168
32bitMAC 付き暗号化	2769.5	2867	3174	3788	5016	7472

■ALE　ALE は FSE 2013 で Rijmen らによって提案されたアルゴリズムである。AES-NI を積極的に利用することが可能な設計が採られている。認証暗号専用 (Dedicated) の設計ではあるが、モードの設計にも近く、性能比較も AES のモードとの比較をしている。文献 [2] に示される AES の Serial 実装（S-box 1 つを使いまわして暗号化演算を行う HW 実装）をベースにした 65nm CMOS スタンダードセルライブラリによる実装評価結果を表 2.23 に纏める。

表 2.23　ALE の回路性能

Design	Area[gate]	Clock cycles / block	Overhead cycles / message	Power [uW]
AES-ECB	2,435	226	-	87.84
AES-OCB2	4,612	226	452	171.23
AES-OCB2 e/d	5,916	226	452	211.01
ASC-1 A	4,793	370	904	169.11
ASC-1 A e/d	4,964	370	904	193.71
ASC-1 B	5,517	235	904	199.02
ASC-1 B e/d	5,632	235	904	207.13
AES-CCM	3,472	452	-	128.31
AES-CCM e/d	3,765	452	-	162.15
ALE	2,579	105	678	94.87
ALE e/d	2,700	105	678	102.32

ここで、ASC-1 は文献 [2] で示されるアルゴリズムであり、ALE の原型と呼べるアルゴリズムである。ALE は AES-OCB2 に対して半分の回路規模で 2 倍の処理速度が得られる。

図 2.2 に文献 [2] に記載される Sandy Bridge (AES-NI) 利用時のソフトウェア性能を示す。図から ALE は AES-OCB3 と同程度の処理性能を持つことがわかる。

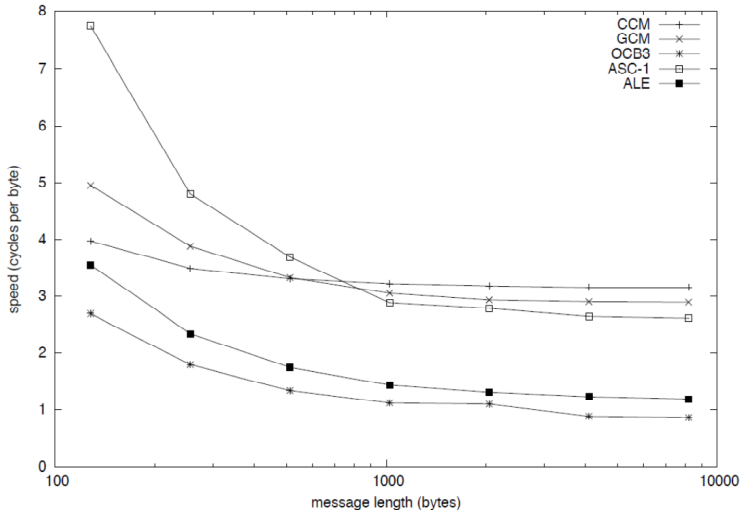


図 2.2　ALE のソフトウェア性能

■FIDES　FIDES は CHES 2013 で提案された認証暗号であり、以下のような特徴を持つ。

・ 論理回路として 793 gate で実装可能
・ Sponge 構造で 5bit、 6bit S-box を持つ
・ 鍵長、ステートが 80bit、160bit と 96bit、192bit の 2 種類ある

表 2.24 に文献 [3] に記載されるハードウェア性能を示す。文献 [3] では 3 種類の CMOS プロセスを用いた評価結果を示している。表中、Threshold implementation とはハードウェアにおけるサイドチャネル対策のための一方式を指す。

表 2.24　FIDES のハードウェア性能

Design	Security (bits)	Area (GE)	Frequency (kHz)	Latency	Throughput (kb/s)	Power (μW)
	Advanced NXP 90 nm CMOS process, typical PVT (25 °C, 1.2 V)					
FIDES-80-S	80	793	100	47	10.64	N/A
FIDES-80-4S	80	1178	100	23	21.74	N/A
FIDES-80-R	80	2922	100	1	500.00	N/A
FIDES-80-T	80	2876	100	47	10.64	N/A
FIDES-96-S	96	1001	100	47	12.77	N/A
FIDES-96-4S	96	1305	100	23	26.09	N/A
FIDES-96-R	96	6673	100	1	600.00	N/A
FIDES-96-T	96	4792	100	47	12.77	N/A
	NANGATE 45 nm CMOS process, typical PVT (25 °C, 1.1 V)					
FIDES-80-S	80	1244	100	47	10.64	N/A
FIDES-80-4S	80	1819	100	23	21.74	N/A
FIDES-80-R	80	4023	100	1	500.00	N/A
FIDES-80-T	80	4696	100	47	10.64	N/A
FIDES-96-S	96	1584	100	47	12.77	N/A
FIDES-96-4S	96	2023	100	23	26.09	N/A
FIDES-96-R	96	9180	100	1	600.00	N/A
FIDES-96-T	96	7541	100	47	12.77	N/A
	UMC 130 nm CMOS process, typical PVT (25 °C, 1.2 V)					
FIDES-80-S	80	1153	100	47	10.64	1.97
FIDES-80-4S	80	1682	100	23	21.74	2.82
FIDES-80-R	80	4175	100	1	500.00	7.90
FIDES-80-T	80	4267	100	47	10.64	7.47
FIDES-96-S	96	1453	100	47	12.77	2.49
FIDES-96-4S	96	1870	100	23	26.09	3.12
FIDES-96-R	96	8340	100	1	600.00	14.82
FIDES-96-T	96	6812	100	47	12.77	11.84

Fides-xy-S : Serial architecture (1 S-box).

Fides-xy-4S : Architecture with 4 S-boxes.

Fides-xy-R : Round-based architecture (32 S-boxes).

Fides-xy-T : Threshold implementation (1 S-box).

■Phelix　Phelix は 2004 年に eSTREAM に提案された MAC 付きストリーム暗号である。Phase 2 で落選で落選している。表 2.25 に文献 [4] に記載されるソフトウェア性能を示す。文献 [4] では Pentium M CPU での速度性能が示されている。

表 2.25　Phelix のソフトウェア性能

Operation	Version	Packet Size (N)			Approximate
		64 bytes	256 bytes	1024 bytes	Equation (clks)
Encrypt	C	41.6 cpb	20.3 cpb	15.0 cpb	$1810 + 13.2N$
Decrypt	C	42.3 cpb	21.1 cpb	15.8 cpb	$1610 + 14.0N$
Encrypt	ASM	18.5 cpb	9.8 cpb	7.4 cpb	$810 + 6.6N$
Decrypt	ASM	18.2 cpb	9.6 cpb	7.4 cpb	$750 + 6.7N$

cbp: clocks per byte

■Mode of Operation　AES-NI 前提で CCM、GCM、OCB3 など認証暗号用のモードに対する速度性能評価が文献 [5, 6, 7, 8] などで実施されている。表 2.26 にそれぞれの評価結果をまとめる。

表 2.26　暗号利用モードのソフトウェア性能 (Sandy Bridge)

Mode	cbp	data	Source
ECB	0.702	4KB	[6]
	0.853	8KB	OpenSSL 1.0.1c
CTR	0.691	4KB	[6]
	0.79	16KB	[RWC2013]
	0.916	8KB	OpenSSL 1.0.1c
OCB2	1.016	4KB	[6] (連続 2 倍)
	1.350	4KB	[6] (通常 2 倍)
OCB3	0.818	4KB	[6]
	0.87	4KB	[9]
GCM	2.47	16KB	[7]
	2.53	4KB	[7]
	2.564	4KB	[6]
	2.899	8KB	OpenSSL 1.0.1c

■CAESAR プロジェクト提案暗号　認証暗号アルゴリズムの公募プロジェクトである CAESAR プロジェクトへ提案されているアルゴリズムについて、提案者らが提示している実装性能を以下に示す。なお、既に鍵の全数探索よりも効率のよい攻撃方法が見つかっているアルゴリズムを含め、まとめている点に留意されたい。

表 2.27 に、FPGA の性能評価結果をまとめる。3 つのアルゴリズムで性能値が示されている。

表 2.28 に、ASIC の性能評価結果をまとめる。5 つのアルゴリズムで性能値が示されている。表 2.29 は、具体的な実装結果ではないが、ゲート規模の見積もりなどを実施しているアルゴリズムの性能値をまとめた結果である。5 つのアルゴリズムで性能値が示されている。

表 2.30 に、Mode of operation の提案で、Ivy Bridge マイクロアーキテクチャを実装ターゲットとしたソフトウェアの性能評価結果をまとめる。3 つのアルゴリズムで性能値が示されている。同様に表 2.31 には Dedicated としての提案に対する性能値をまとめる。

表 2.27　CAESAR 候補の FPGA 性能 (実装値)

Algorithm	Platform	Area	Freq. (MHz)	Throughput (Mbps)	Source
ICEPOLE	Xilinx Virtex6	1501 (slices/ALUT)	N/A	41,364	[21]
	Altera Stratix IV	4564 (slices/ALUT)	N/A	38,779	[21]
KIASU-BC	Xilinx Virtex5	1989 (slices)	N/A	1,080	[24]
pi-Cipher	Xilinx Virtex6	41 (slices)	N/A	N/A	[32]

表 2.28　CAESAR 候補の ASIC 性能 (実装値)

Algorithm	Area	Freq. (MHz)	Throughput (Mbps)	Source
CLOC	17137.75 (GE)	100	685.71	[17]
Minalpher-P	2810 (GE)			
NORX	62000 (GE)	125	10240	[30]
SCREAM-10 (Enc-/Dec-only)*1	12,951 (μm^2)	751	4577	[36]
SCREAM-10 (Enc-/Dec-only)*2	17,292 (μm^2)	446	5190	[36]
SCREAM-10 (Enc+Dec)*1	17,292 (μm^2)	751	4577	[36]
SCREAM-10 (Enc+Dec)*2	25,974 (μm^2)	446	5190	[36]
iSCREAM-12 (Enc-/Dec-only)*1	13,375 (μm^2)	740	3789	[36]
iSCREAM-12 (Enc-/Dec-only)*2	17,024 (μm^2)	448	4411	[36]
iSCREAM-12 (Enc+Dec)*1	13,375 (μm^2)	740	3789	[36]
iSCREAM-12 (Enc+Dec)*2	17,024 (μm^2)	448	4411	[36]
SILC	15675.5 (GE)	100	764.12	[38]

*1 1 round per cycle

*2 2 rounds per cycle

60

表 2.29　CAESAR 候補の ASIC 性能 (概算値)

Algorithm	Area (GE)	Source
Deoxys-BC-128-128	3400	[18]
Deoxys-BC-256-128	4400	[18]
Deoxys-128-128	4600	[18]
Deoxys-128-128	5600	[18]
Joltik$^{\neq}$-64-64	2100	[19]
Joltik$^{\neq}$-80-48	2100	[19]
Joltik$^{\neq}$-96-96	2600	[19]
Joltik$^{\neq}$-128-64	2600	[19]
Joltik$^{=}$-64-64	2600	[19]
Joltik$^{=}$-80-48	2600	[19]
Joltik$^{=}$-96-96	3100	[19]
Joltik$^{=}$-128-64	3100	[19]
KIASU$^{\neq}$	4000	[23]
KIASU$^{=}$	5000	[23]
LAC	1300	[25]
Sablier	1925	[35]

表 2.30　CAESAR 候補 (Mode of operation) のソフトウェア性能 (Ivy Bridge)

Algorithm	Speed (cpb)	Message length (bytes)	Source
AES-CPFB (Enc)	2	1500	[13]
	1.47	32768	[13]
AES-CPFB (Dec)	7.5	1500+	[13]
AES-SILC	4.9	long	[38]
PRESENT-SILC	42	long	[38]
LED-SILC	40	long	[38]
Scream-10	7.1	long	[36]
iScream-12	9.1	long	[36]

表 2.31　CAESAR 候補 (Dedicated) のソフトウェア性能 (Ivy Bridge)

Algorithm	Speed (cpb)	Message length (bytes)	Source
ICEPOLE (without special instruction sets)	9	N/A	[22]
Minalpher	23.1	31	[27]
	14.4	8192	[27]
	14.4	65536	[27]

表 2.32 に、Mode of operation の提案で、Sandy Bridge マイクロアーキテクチャを実装ターゲットとしたソフトウェアの性能評価結果をまとめる。5 つのアルゴリズムで性能値が示されている。同様に表 2.33 には Dedicated としての提案に対する性能値をまとめる。

表 2.32: CAESAR 候補 (Mode of operation) のソフトウェア性能 (Sandy Bridge)

Algorithm	Speed (cpb)	Message length (bytes)	Source
AES-JAMBU	17.7	64	[14]
	14.54	128	[14]
	13.06	256	[14]
	12.27	512	[14]
	11.86	1024	[14]
	11.60	4096	[14]
AEGIS-128L(Enc/Dec)	3.68/3.81	64	[11]
	2.05/2.12	128	[11]
	1.23/1.27	256	[11]
	0.83/0.85	512	[11]
	0.63/0.63	1024	[11]
	0.48/0.48	4096	[11]
AEGIS-128(Enc/Dec)	3.37/3.78	64	[11]
	1.99/2.17	128	[11]
	1.30/1.36	256	[11]
	0.96/1.02	512	[11]
	0.80/0.84	1024	[11]
	0.66/0.67	4096	[11]
AEGIS-256(Enc/Dec)	3.51/4.00	64	[11]
	2.10/2.35	128	[11]
	1.34/1.51	256	[11]
	1.03/1.09	512	[11]
	0.86/0.90	1024	[11]
	0.70/0.74	4096	[11]
Deoxys$^{\neq}$-128-128	2.30	128	[18]
	1.73	256	[18]
	1.45	512	[18]
	1.36	1024	[18]
	1.15	2048	[18]
	1.13	4096	[18]
Deoxys$^{\neq}$-256-128	4.26	128	[18]
	2.53	256	[18]

62

Algorithm	Speed (cpb)	Message length (bytes)	Source
	1.92	512	[18]
	1.57	1024	[18]
	1.48	2048	[18]
	1.32	4096	[18]
Deoxys$^{=}$-128-128	4.50	128	[18]
	3.42	256	[18]
	2.84	512	[18]
	2.61	1024	[18]
	2.43	2048	[18]
	2.33	4096	[18]
Deoxys$^{=}$-256-128	7.89	128	[18]
	5.13	256	[18]
	3.55	512	[18]
	3.07	1024	[18]
	2.75	2048	[18]
	2.59	4096	[18]
KIASU$^{\neq}$	1.02	4096	[23]
KIASU$^{=}$	1.98	4096	[23]
Tiaoxin	2.49	128	[41]
	1.45	256	[41]
	0.91	512	[41]
	0.65	1024	[41]
	0.50	2048	[41]
	0.44	4096	[41]
	0.40	8192	[41]
	0.38	2^{16}	[41]

表 2.33　CAESAR 候補 (Dedicated) のソフトウェア性能 (Sandy Bridge)

Algorithm	Speed (cpb)	Message length (bytes)	Source
ACORN	72.1	64	[10]
	41.5	128	[10]
	26.3	256	[10]
	18.6	512	[10]
	14.7	1024	[10]
	12.8	2048	[10]
	11.9	4096	[10]

表 2.34 に、Mode of operation の提案で、Haswell マイクロアーキテクチャを実装ターゲットとしたソフトウェアの性能評価結果をまとめる。9 つのアルゴリズムで性能値が示されている。同様に表 2.35 に Dedicated としての提案に対する性能値をまとめる。

表 2.34: CAESAR 候補 (Mode of operation) のソフトウェア性能 (Haswell)

Algorithm	Speed (cpb)	Message length (bytes)	Source
AES-COPA	1.44	128(short)	[12]
	1.29	2048(long)	[12]
AEZ	0.38(検証失敗時)	1500	[15]
	0.89	1500	[15]
	0.72	16384	[15]
AEGIS-128L(Enc/Dec)	3.44/3.45	64	[11]
	1.88/1.88	128	[11]
	1.11/1.09	256	[11]
	0.71/0.70	512	[11]
	0.51/0.50	1024	[11]
	0.37/0.35	4096	[11]
AEGIS-128(Enc/Dec)	3.29/2.98	64	[11]
	1.92/1.77	128	[11]
	1.24/1.16	256	[11]
	0.91/0.86	512	[11]
	0.73/0.81	1024	[11]
	0.61/0.60	4096	[11]
AEGIS-256(Enc/Dec)	3.98/3.88	64	[11]
	2.28/2.22	128	[11]
	1.42/1.39	256	[11]

64

Algorithm	Speed (cpb)	Message length (bytes)	Source
	0.99/0.98	512	[11]
	0.78/0.77	1024	[11]
	0.62/0.62	4096	[11]
Deoxys$^{\neq}$-128-128	2.25	128	[18]
	1.84	256	[18]
	1.64	512	[18]
	1.55	1024	[18]
	1.49	2048	[18]
	1.46	4096	[18]
Deoxys$^{\neq}$-256-128	3.68	128	[18]
	2.66	256	[18]
	2.14	512	[18]
	1.88	1024	[18]
	1.76	2048	[18]
	1.69	4096	[18]
Deoxys$^{=}$-128-128	4.07	128	[18]
	3.43	256	[18]
	3.12	512	[18]
	2.97	1024	[18]
	2.89	2048	[18]
	2.85	4096	[18]
Deoxys$^{=}$-256-128	5.68	128	[18]
	4.44	256	[18]
	3.82	512	[18]
	3.51	1024	[18]
	3.36	2048	[18]
	3.28	4096	[18]
HS1-SIV	0.8	N/A	[20]
KIASU$^{\neq}$	0.74	4096	[23]
KIASU$^{=}$	1.39	4096	[23]
Marble	1.6	8192	[26]
Silver(Enc/Dec)(AES-NI)	10.8/9.6	44	[39]
	1/1.2	1536	[39]
	0.73/0.81	long	[39]
Silver(Enc/Dec)(non-AES-NI)	30.4/28.2	44	[39]
	11.85/13.59	1536	[39]
	11.45/12.9	long	[39]
Tiaoxin	0.31	8192	[41]

Algorithm	Speed (cpb)	Message length (bytes)	Source
	0.28	long	[41]

表 2.35　CAESAR 候補 (Dedicated) のソフトウェア性能 (Haswell)

Algorithm	Speed (cpb)	Message length (bytes)	Source
ICEPOLE (without special instruction sets)	8	N/A	[22]
Minalpher	5.76	8192	[28]
MORUS-640(Enc/Dec)	7.72/7.99	64	[29]
	1.18/1.23	4096	[29]
	1.11/1.16	long	[29]
MORUS-1280(Enc/Dec)	8.28/8.46	64	[29]
	0.78/0.80	4096	[29]
	0.69/0.69	long	[29]
NORX64-6-1(Ref/AVX2)*3	1248.00/748.24	8	[30]
	156.61/93.23	64	[30]
	9.85/5.71	576	[30]
	7.77/4.47	1536	[30]
	7.00/3.98	4096	[30]
	6.63/3.73	long	[30]
NORX64-4-1(Ref/AVX2)*3	863.12/509.51	8	[30]
	106.94/63.38	64	[30]
	6.71/3.83	576	[30]
	5.27/3.01	1536	[30]
	4.76/2.66	4096	[30]
	4.50/2.51	long	[30]

*3 Ref: 移植可能な C レファレンス実装、AVX2: AVX2 利用の最適実装

最後に表 2.36 として、上記のいずれの分類にも含まれない候補のソフトウェアの性能評価結果をまとめる。

表 2.36: CAESAR 候補のソフトウェア性能 (Others)

Algorithm	Platform	ROM/RAM (bytes)	Speed (cpb)	Message length (bytes)	Source
HS1-SIV	MIPS32	N/A	16	N/A	[20]
	Cortex-A9	N/A	5	N/A	[20]

Algorithm	Platform	ROM/RAM (bytes)	Speed (cpb)	Message length (bytes)	Source
LAC	Core i7-3612QM	N/A	720	12	[25]
			589	16	[25]
			440	32	[25]
			256	64	[25]
			206	128	[25]
			174	256	[25]
			152	512	[25]
			144	1024	[25]
			140	2048	[25]
			138	4096	[25]
Minalpher	RL78	1275/470	514	long	[27]
NORX32-6-1 (Ref/NEON)*4	Samsung Exynos 4412 Prime (Cortex-A9)	N/A	794.12/541.00	8	[30]
			128.66/77.78	64	[30]
			42.14/22.79	576	[30]
			35.45/18.36	1536	[30]
			32.35/16.70	4096	[30]
			31.56/15.66	long	[30]
NORX32-4-1 (Ref/NEON)*4	Samsung Exynos 4412 Prime (Cortex-A9)	N/A	663.75/434.88	8	[30]
			97.94/61.73	64	[30]
			30.50/16.40	576	[30]
			24.94/12.77	1536	[30]
			22.86/11.41	4096	[30]
			21.57/10.57	long	[30]
NORX64-6-1 (Ref/AVX)*5	Core i7-2630QM	N/A	304.00/198.00	8	[30]
			37.75/24.81	64	[30]
			11.54/7.52	576	[30]
			9.08/5.90	1536	[30]
			8.14/5.24	4096	[30]
			7.69/4.94	long	[30]
NORX64-4-1 (Ref/AVX)*5	Core i7-2630QM	N/A	208.00/133.50	8	[30]
			26.00/16.69	64	[30]
			7.94/5.03	576	[30]
			6.24/3.91	1536	[30]
			5.59/3.49	4096	[30]
			5.28/3.28	long	[30]
NORX64-6-1 (Ref/AVX)*5	Core i7-3667U	N/A	371.50/276.00	8	[30]
			34.87/25.44	64	[30]

Algorithm	Platform	ROM/RAM (bytes)	Speed (cpb)	Message length (bytes)	Source
			10.59/7.71	576	[30]
			8.32/6.03	1536	[30]
			7.46/5.37	4096	[30]
			7.04/5.04	lonog	[30]
NORX64-4-1	Core i7-3667U	N/A	310.00/218.00	8	[30]
(Ref/AVX)[*5]			24.93/17.18	64	[30]
			7.43/5.16	576	[30]
			5.86/4.01	1536	[30]
			5.24/3.59	4096	[30]
			4.92/3.37	long	[30]
POET	Core i5-4300U	N/A	4.61	128	[34]
			4.24	256	[34]
			4.13	512	[34]
			4.02	1024	[34]
			3.92	2048	[34]
OMD-SHA256	Core i5-2415M	N/A	44.56	128	[31]
			28.77	4096	[31]
OMD-SHA512	Core i5-2415M	N/A	45.93	128	[31]
			23.28	4096	[31]
Scream-10[*6]	Cortex A15	N/A	21.8	long	[36]
	Atom Cedarview	N/A	55	long	[36]
	Core i7 Nehalem	N/A	9.3	long	[36]
	Atmel AVR	3221/80	7646(E)/7672(D)	N/A	[36]
	Atmel AVR	1723/80(Enc-only)	7646	N/A	[36]
	Atmel AVR	1751/80(Dec-only)	7672	N/A	[36]
iScream-12[*6]	Cortex A15	N/A	26.2	long	[36]
	Atom Cedarview	N/A	65	long	[36]
	Core i7 Nehalem	N/A	11.2	long	[36]
	Atmel AVR	1975/64	8724(E)/8724(D)	long	[36]
	Atmel AVR	1595/64(Enc-only)	8724	N/A	[36]
	Atmel AVR	1593/64(Dec-only)	8724	N/A	[36]
STRIBOB	Core i7 860	N/A	25.3	N/A	[40]

[*4] Ref: 移植可能な C レファレンス実装、NEON: NEON 利用の最適実装

[*5] Ref: 移植可能な C レファレンス実装、AVX: AVX 利用の最適実装

[*6] tweakable block cipher のみの実装

2.6.2.2　まとめ

本節では、主要な認証暗号の実装性能 (ハードウェア、ソフトウェア) 調査結果をまとめた。本調査は CAESAR プロジェクトがスタートし、第二ラウンド進出アルゴリズムを選定している段階で実施しているため、数多くのアルゴリズムについて性能値を掲載している。しかしながら、これらはあくまで著者らの主張に基づいた提示であり、本資料記載のデータを用いてアルゴリズム間の比較を行う目的にはそぐわないことに注意されたい。

今後、安全性やサイドチャネル対策との関連性を含めプロジェクトでの絞り込みについて動向を注視していく必要があると考える。

参考文献

[1] Martin Ågren, Martin Hell, Thomas Johansson and Willi Meier: Grain-128a: a new version of Grain-128 with optional authentication. IJWMC 5(1): 48-59, 2011.

[2] Andrey Bogdanov, Florian Mendel, Francesco Regazzoni, Vincent Rijmen and Elmar Tischhauser: ALE: AES-Based Lightweight Authenticated Encryption. FSE2013.

[3] Begül Bilgin, Andrey Bogdanov, Miroslav Knezevic, Florian Mendel and Qingju Wang: FIDES: Lightweight Authenticated Cipher with Side-Channel Resistance for Constrained Hardware. CHES2013.

[4] Doug Whiting, Bruce Schneier, Stefan Lucks and Frédéric Muller: Phelix Fast Encryption and Authentication in a Single Cryptographic Primitive. https://www.schneier.com/paper-phelix.pdf

[5] Kazumaro Aoki, Tetsu Iwata and Kan Yasuda: How Fast Can a Two-Pass Mode Go? A Parallel Deterministic Authenticated Encryption Mode for AES-NI. DIAC 2012

[6] Kazumaro Aoki: Optimization of mode implementations on Sandy Bridge. SCIS 2013

[7] Shay Gueron: AES-GCM for Efficient Authenticated Encryption - Ending the Reign of HMAC-SHA-1? https://crypto.stanford.edu/RealWorldCrypto/slides/gueron.pdf

[8] Shay Gueron: AES-GCM software performance on the current high end CPUs as a performance baseline for CAESAR competition? http://2013.diac.cr.yp.to/slides/gueron.pdf

[9] Phillip Rogaway, Mihir Bellare, John Black, Ted Krovetz, and Tom Shrimpton: The Evolution of Authenticated Encryption.
http://hyperelliptic.org/DIAC/slides/sweden-rogaway-ae-2012b.pdf

[10] Hongjun Wu, "ACORN: A Lightweight Authenticated Cipher (v1),"
http://competitions.cr.yp.to/round1/acornv1.pdf

[11] Hongjun Wu, Bart Preneel, "AEGIS: A Fast Authenticated Encryption Algorithm (v1),"
http://competitions.cr.yp.to/round1/aegisv1.pdf

[12] Elena Andreeva, Andrey Bogdanov, Atul Luykx, Bart Mennink, Elmar Tischhauser, Kan Yasuda, "AES-COPA v.1," http://competitions.cr.yp.to/round1/aescopav1.pdf

[13] Miguel Montes, Daniel Penazzi, "AES-CPFB v1,"
http://competitions.cr.yp.to/round1/aescpfbv1.pdf

[14] Hongjun Wu, Tao Huang, "JAMBU Lightweight Authenticated Encryption Mode and AES-JAMBU (v1),"
http://competitions.cr.yp.to/round1/aesjambuv1.pdf

[15] Viet Tung Hoang, Ted Krovetz, Phillip Rogaway, "AEZ v3: Authenticated Encryption by Enciphering,"
http://web.cs.ucdavis.edu/~rogaway/aez/aez.pdf

[16] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, "CLOC: Compact Low-Overhead CFB,"

http://competitions.cr.yp.to/round1/clocv1.pdf

[17] Tetsu Iwata, "CAESAR candidate SILC," http://2014.diac.cr.yp.to/slides/iwata-silc.pdf

[18] Jérémy Jean, Ivica Nikolić, Thomas Peyrin, "Deoxys v1," http://competitions.cr.yp.to/round1/deoxysv1.pdf

[19] Jérémy Jean, Ivica Nikolić, Thomas Peyrin, "Joltik v1," http://competitions.cr.yp.to/round1/joltikv1.pdf

[20] Ted Krovetz, "HS1-SIV," http://2014.diac.cr.yp.to/slides/krovetz-hs1.pdf

[21] Pawel Morawiecki, Kris Gaj, Ekawat Homsirikamol, Krystian Matusiewicz, Josef Pieprzyk, Marcin Rogawski, Marian Srebrny, Marcin Wójcik, "ICEPOLE v1," http://competitions.cr.yp.to/round1/icepolev1.pdf

[22] Marcin Rogawski, "CAESAR candidate ICEPOLE", http://2014.diac.cr.yp.to/slides/rogawski-icepole.pdf

[23] Jérémy Jean, Ivica Nikolić, Thomas Peyrin, "KIASU v1," http://competitions.cr.yp.to/round1/kiasuv1.pdf

[24] Thomas Peyrin, "CAESAR candidate KIASU," http://competitions.cr.yp.to/round1/kiasuv1.pdf

[25] Lei Zhang, Wenling Wu, Yanfeng Wang, Shengbao Wu, Jian Zhang, "LAC: A Lightweight Authenticated Encryption Cipher," http://competitions.cr.yp.to/round1/lacv1.pdf

[26] Jian Guo, "Marble Specification Version 1.1," http://competitions.cr.yp.to/round1/marblev11.pdf

[27] Yu Sasaki, Yosuke Todo, Kazumaro Aoki, Yusuke Naito, Takeshi Sugawara, Yumiko Murakami, Mitsuru Matsui, Shoichi Hirose, "Minalpher v1," http://competitions.cr.yp.to/round1/minalpherv1.pdf

[28] Kazumaro Aoki, "Observations on Prøst and Minalpher," https://www.cryptolux.org/mediawiki-esc2015/images/c/cb/Slide.pdf

[29] Hongjun Wu, Tao Huang, "The Authenticated Cipher MORUS (v1)," http://competitions.cr.yp.to/round1/morusv1.pdf

[30] Jean-Philippe Aumasson, Philipp Jovanovic, Samuel Neves, "NORX v1," http://competitions.cr.yp.to/round1/norxv1.pdf

[31] Simon Cogliani, Diana-Ştefania Maimuţ, David Naccache, Rodrigo Portella do Canto, Reza Reyhanitabar, Serge Vaudenay, Damian Vizár, "OMD A Compression Function Mode of Operation for Authenticated Encryption," http://2014.diac.cr.yp.to/slides/reyhanitabar-omd.pdf

[32] Danilo Gligoroski, Hristina Mihajloska, Simona Samardjiska, Håkon Jacobsen, Mohamed El-Hadedy, Rune Erlend Jensen, "π-Cipher v1," http://competitions.cr.yp.to/round1/picipherv1.pdf

[33] Danilo Gligoroski, "CAESAR candidate PiCipher," http://2014.diac.cr.yp.to/slides/gligoroski-picipher.pdf

[34] Farzaneh Abed, Scott Fluhrer, John Foley, Christian Forler, Eik List, Stefan Lucks, David McGrew, Jakob Wenzel, "The POET Family of On-Line Authenticated Encryption Schemes," http://competitions.cr.yp.to/round1/poetv101.pdf

[35] Bin Zhang, Zhenqing Shi, Chao Xu, Yuan Yao, Zhenqi Li, "Sablier v1," http://competitions.cr.yp.to/round1/sablierv1.pdf

[36] Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, Kerem Varici, François Durvaux, Lubos Gaspar, Stéphanie Kerckhof, "SCREAM & iSCREAM Side-Channel Resistant Authenticated Encryption with

Masking," http://competitions.cr.yp.to/round1/screamv1.pdf

[37] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, Eita Kobayashi, "SILC: SImple Lightweight CFB," http://competitions.cr.yp.to/round1/silcv1.pdf

[38] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, Eita Kobayashi, "SILC: SImple Lightweight CFB," http://2014.diac.cr.yp.to/slides/iwata-silc.pdf

[39] Daniel Penazzi, Miguel Montes, "Silver and AESCPFB," http://2014.diac.cr.yp.to/slides/penazzi-silver-cpfb.pdf

[40] Markku-Juhani O. Saarinen, "The STRIBOBr1 Authenticated Encryption Algorithm," http://competitions.cr.yp.to/round1/stribobr1.pdf

[41] Ivica Nikolić, "Tiaoxin–346," http://competitions.cr.yp.to/round1/tiaoxinv1.pdf

第 3 章

軽量暗号に関する現状調査: 軽量暗号に関わる新しい技術動向

3 章の執筆担当者は下記の通りである.

第 3 章　軽量暗号に関わる新しい技術動向	
3.1 章　低レイテンシ暗号	崎山委員
3.2 章　サイドチャネル攻撃耐性	成吉委員
3.3 章　CAESAR プロジェクト	岩田委員
3.4 章　軽量暗号の活用事例および標準化動向	小川委員

3.1　低レイテンシ暗号

3.1.1　はじめに

低レイテンシ暗号 (Low-Latency Cryptography) に関する論文のうち、特に欧州で研究が活発であるブロック暗号を用いた Low-Latency Encryption/Decryption について技術動向調査を行った。ハードウェア実装に関する論文 [1, 2, 3] を紹介し、今後の展望について述べる。

3.1.2　Low-Latency Cryptography 研究のモチベーション

暗号処理における低レイテンシ性は、暗号処理時の応答速度を重視するデータ通信アプリケーションに求められている。例えば、車の自動運転支援システム（Car2X communication)、セキュア・ストレージ及び CPU と外部ストレージ間のデータを暗号化するバス・エンクリプションである。半導体加工技術の高精度化（CMOS プロセスのの微細化）による集積回路の信号遅延時間短縮が大きく期待できない中、低レイテンシ暗号を実現するためには、暗号処理に要する計算量自体を大幅に削減する必要がある。これが、軽量暗号が新たに求められる理由のひとつと考える。現在広く使われている AES ブロック暗号では、回路規模、レイテンシともに上述のようなアプリケーションが求める性能要求を満たさない。例えば、1〜2 ns のレイテンシ性能を実現する AES 暗号ハードウェアは、現在の回路技術では実装が困難である。

3.1.3 ブロック暗号による Low-Latency Encryption/Decryption の性能評価

Knežević らによって CHES2012 で発表された論文 [1] では、暗号処理回路を 1 サイクルあるいは 2 サイクルでが完了するように実装し、数 10 MHz〜数 100 MHz のオーダーの最大動作周波数での処理時間をレイテンシとしている。つまり、レイテンシは数 ns〜数 10ns 程度となる。本報告では簡単のために、1 サイクルで処理が完了する場合についてのみ紹介する。90 nm CMOS テクノロジで合成した場合、 AES-128 のレイテンシは 14.8 ns、mCrypton-128 では 9.7 ns、PRESENT-128 では 14.3 ns と報告されている。Encryption/Decryption 両機能を搭載した場合、 AES-128 のレイテンシは約 17.8ns となり、性能の低下が見られるが、mCrypton-128 と PRESENT-128 ではそれぞれ 9.8 ns と 14.8 ns でとなる、ほとんど差異がないと評価されている。 3 つの暗号方式それぞれの回路規模は、AES-128, mCrypton-128, PRESENT-128 の順に、約 360 kGE、50 kGE、80 kGE（GE: Gate Equivalent の略、回路面積を表す単位）である。この結果から、mCrypton-128 が優れているように見えるが、安全性を犠牲にしている可能性がある。また、低レイテンシ暗号の場合には、回路規模はそれほど重要ではなく、むしろレイテンシに重きを置いた評価が好ましいと思われる。

Borghoff らによる ASIACRYPT2012 の発表論文 [2] で、低レイテンシのブロック暗号 PRINCE が提案された。4 ビット S-box による非線形演算と線形演算で構成されるデータ・パスは 64 ビット長で、鍵は 128 ビット長である。AES の鍵スケジュールと比べて、非常に単純な鍵スケジュール方式を採用している。回路規模は、約 8 kGE と報告されている。レイテンシは、45 nm CMOS テクノロジで 4.7 ns、90 nm CMOS テクノロジで 13.9 ns と報告されている。

SCIS2014 で、鈴木らは PRESENT と PRINCE の低レイテンシ実装を発表した [3]。PRESENT と PRINCE を 45 nm CMOS テクノロジで合成した結果、回路規模はそれぞれ 22 kGE と 8 kGE となり、レイテンシは 9.03 ns と 5.49 ns となった。ちなみに AES では 174 kGE で 12.25ns のレイテンシであった。ただし、以上の回路規模の数値は、暗号処理回路のみに基づくものであり、ARM プロセッサ向けの周辺モジュール用のバス・インターフェイス回路分（AMBA APB: 約 2 kGE）は含まない。この論文 [3] では、RFID タグへの実装に関する興味深い考察が与えられている。RFID タグ・チップのシリコン・ダイのサイズは、基板実装上の制限を受け、300μm 角程度が限界（下限）とされている。CMOS プロセスの微細化にともない、シリコンダイに実装できる回路規模が増大する。例えば 90 nm プロセスでは、300μm 角のシリコン・ダイに 30 kGE のロジック回路が搭載可能である。つまり、PRESENT と PRINCE は 90 nm（より微細な）CMOS テクノロジを用いることで、RFID タグに搭載できる。ただし、パッシブ RFID タグでは、低消費電力が重要となるため、この点は留意する必要がある。

3.1.4 まとめ

ここでは、低レイテンシを実現するいくつかの軽量ブロック暗号に関する技術動向調査を行った。ブロック暗号 PRINCE は、鍵拡張の単純化やデータ・パスの 64 ビット化により、回路規模の低減と低レイテンシ化の両方を同時に実現した。回路規模に対するレイテンシ性能は、アプリケーションによっては十分な性能と言える水準にあると考える。複数ラウンドを 1 サイクルで実装することは、サイドチャネル耐性の向上に繋がることが報告されている [4]。低レイテンシ実装においても同様の耐性向上が期待できるため、今後は、軽量暗号実装における耐タンパー性評価を併せて考える必要があると思われる。

参考文献

[1] Miroslav Knežević, Ventzislav Nikov, Peter Rombouts. Low-Latency Encryption - Is "Lightweight = Light + Wait"? In Emmanuel Prouff and Patrick Schaumont, editors, *Cryptographic Hardware and Embedded Systems — CHES 2012*, volume 7428 of *Lecture Notes in Computer Science*, pages 426-446,, Springer-Verlag, Berlin, Heidelberg, 2012.

[2] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knežević, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and-Tolga Yalçin. PRINCE — A Low-Latency Block Cipher for Pervasive Computing Applications — Extended Abstract. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology — ASIACRYPT 2012 — 18th International Conference on the Theory and Application of Cryptology and Information Security*, volume 7658 of *Lecture Notes in Computer Science*, pages 208-225, Springer-Verlag, Berlin, Heidelberg, 2012

[3] 鈴木大輔, 菅原健, 佐伯稔. 軽量/低遅延暗号のハードウェア実装性能について. *2014* 年 暗号と情報セキュリティシンポジウム — *SCIS 2014*, 6 pages, 2014.

[4] Shivam Bhasin, Sylvain Guilley, Laurent Sauvage, Jean-Luc Danger, Unrolling Cryptographic Circuits, Unrolling Cryptographic Circuits: A Simple Countermeasure Against Side-Channel Attacks. In Josef Pieprzyk, editor, *Topics in Cryptology — CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010*, volume 5985 of *Lecture Notes in Computer Science*, pp.195-207, Springer-Verlag, Berlin, Heidelberg, 2010.

3.2 サイドチャネル攻撃耐性

本章では、軽量暗号技術に関する現状調査のうち、サイドチャネル攻撃耐性に関する文献調査結果を記載する。

3.2.1 調査対象

サイドチャネル攻撃耐性に関して、2012 年度まで CRYPTREC 暗号実装委員会にて活動していたサイドチャネルワーキンググループ活動の報告書 [1]、ならびにセキュリティ認証に関する規格 ISO/IEC15408 のコモンクライテリア承認アレンジメント (Common Criteria Recognition Arrangement) の web サイトに登録されている攻撃手法 [2] を参考に、以下 (i)(ii) を調査対象とし、調査結果を以降に記す。

(i) サイドチャネル攻撃 (リーク解析、電流解析。電磁波解析も含む)
(ii) 故障利用攻撃

物理攻撃、例えば文献 [2] には IC へのアクセスもしくは加工をおこなう物理解析などの記載はあるが、暗号アルゴリズムによる対策等の記載がなかったため、各軽量暗号アルゴリズムにおける物理攻撃耐性の評価は調査対象から除外した。

CRYPTREC Report 2012 暗号実装委員会報告ならびに文献 [2] に記載されていない攻撃手法、例えば文献 [3] は AES を攻撃対象として鍵抽出を試みているが、これらも調査対象外とした。調査対象とする暗号技術は、ブロック暗号のうち CRYPTREC 電子政府推奨暗号の AES、 TDES、 Camellia、ISO/IEC 29192-2 記載の PRESENT[4]、CLEFIA、ならびに LED[5]、 Piccolo[6]、 TWINE[7]、 PRINCE[8] とした。

3.2.2 軽量暗号アルゴリズムにおけるサイドチャネル攻撃 (リーク解析) の耐性調査

現状調査として、リーク解析 [9] における各種解析手法に関する文献、リーク解析に関する測定手法に関する文献、軽量暗号に関するリーク解析ならびに耐性を題目とした文献、リーク解析対策の最新手法、リーク耐性全般に関する文献について順に報告する。

3.2.2.1 リーク解析における各種解析手法の現状調査

ブロック暗号を対象としたリーク解析において、差分電力解析 (DPA) ならびにその派生の解析方法 [10] を示す。

1. DPA。演算途中の特定の 1 ビット (選択関数) に着目。入力を変化させたときの各候補鍵で計算した値とリークとの相関を算出し、鍵を推測する手法 [11]
2. CPA。演算途中の特定の値 (複数ビット) に着目。入力を変化させたときの各候補鍵で計算したハミングウエイト、ハミングディスタンス等とリークとの相関を算出し、鍵を推測する手法 [12]
3. High-order 型。演算途中に着目する箇所を複数個所とし、1.、 2. と同様に鍵を推測する手法 [13]
4. 相互情報量を用いた手法 [14]
5. template を用いた攻撃 [10][15]。鍵と入力を変化させて事前に各鍵の電力 (電磁波) のプロファイルを作成し、攻撃時には固定した鍵に対して入力を変化させ、プロファイルとリークとの比較により鍵を推測。
6. シミュレーション結果をリーク結果との相関の入力に与える手法 [16]。攻撃対象となる選択関数の値と、例えば論理シミュレーションでトグル回数を入手しておき、各候補鍵において相関を求めて鍵を推測する手法。

文献 [16] が顕著な例だが、ファンクションから回路を起こした時に発生した AES SBOX での過渡遷移を含んだ消費電力において、SBOX への DPA では 90 万サンプルなのに対して、シミュレーションの結果を相関関数の入力に適用した場合では 13 万サンプルと大幅に減ったとしている。以上から、実装時における過渡遷移の程度など消費電力モデルの精度にばらつきが発生し、プロセスならびに論理合成のコンフィグファイル等で過渡遷移の発生頻度も変わることが文献 [16] から容易に想定できるため、実装結果に関する論文間での厳密な比較は困難である。無対策の軽量暗号アルゴリズムにおいてデータに依存したリークの発生源となり得る演算回路の規模以外の観点からリーク耐性の優劣をつけるのは困難と推測する[*1] 。

3.2.2.2 リーク解析における測定手法の現状調査

リーク解析において電磁波観測を利用した手法 [17] が提案されてから久しいが、2013 年の国際ワークショップ CHES では下記の文献で 2NAND セルに関するリークの違いが報告されており、ゲートレベルですら無対策のものは攻撃されつつあることから、無対策の暗号アルゴリズムでは方式に依存せずに攻撃できるものと類推する。

・On Measurable Side-Channel Leaks inside ASIC Design Primitives[18]

本文献では電磁波リークを観測することでチップ動作を識別する研究がされており、以下の識別が可能とのこと。

- 2NAND セルに対して、入力 (1,1) の状態から入力 (0,0) への変化と入力 (0,1) の変化の区別が可能である (各 1 万波形取得後の平均での比較において)
- メモリのカラム線のアクセスの違いも識別可能

文献 [18] での環境にて鍵抽出評価を実施した場合は、既存の研究結果よりも大幅にサンプル数を減らすことが期待される。

評価環境については評価ボード SASEBO[19] あるいは ZUIHO をキャリブレーションとして使用することで一定の能力の担保はできているものと思われるが、電磁波解析などはコイルから測定場所の選定までパラメータが多く、評価結果に関する論文間の比較は困難である。

3.2.2.3 軽量暗号に関するリーク解析を題目とした文献調査

軽量暗号に関するリーク解析を題目とした発表を文献 [20] にて確認したため、その内容を報告する。本件は特定アルゴリズムに呼応した対策ではない。

文献 [20] においては Adiabatic logics(断熱的回路) を用いた手法でのサイドチャネル攻撃対策がメインである。面積のオーバーヘッドは存在するが、いわゆるグリッチタイプの瞬間的な消費電力の抑制によりサイドチャネル攻撃の耐性が急激に上昇しており、RFID などの低消費電力用途での対策において既存の MDPL[22] や RSL[23] [24] の同じセルレベルでのリーク対策方式と比較して向いているとしている。実チップ評価なし。

その消費電力抑制の効果から軽量暗号のことが触れられている。軽量暗号モジュールは消費電力が比較的小さいことから S/N 比が小さいことが強みである一方で、省電力技術がサイドチャネル攻撃への抵抗を弱めており、まとめとしてサイドチャネル攻撃の成功の可能性を大きくあげており、その実例としてブロック暗号である Keeloq を用いたアプリケーションへの攻撃 [25] を挙げている。

[*1] モジュール内において暗号演算と無関係な回路の動作が多いほど S/N 比が下がるので、そのような暗号アルゴリズムはリーク解析には有利に働く可能性はあると考える。

3.2.2.4　リーク解析対策の最新手法

リーク解析の対策においては秘密情報と秘密情報に依存した消費電力との相関をなくすというのが一般的な手法だが、リーク解析していることを検知することで秘密情報の流出を防ぐ新しいタイプの対策が最近提案されている。

文献 [21] によると、リーク解析のひとつである電磁波解析攻撃の対抗策として EM attack sensor と命名したセンサを暗号モジュールを搭載したチップに実装、実チップによる評価を実施している。EM attack sensor はコイルの形状をしている配線を有しており、その配線に一定の周波数の信号を流しておく。電磁波解析攻撃のため観測用のプローブを近づけると相互インダクタンスが発生し、上記信号の周波数がシフト。この周波数のシフトを観測することで攻撃を受けているかどうかを判別することでリークを防ぐ手法である。

3.2.2.5　リーク解析耐性の文献調査ならびにまとめ

厳密にリーク対策を実施しようとするとセルレベルでの対策、例えば MDPL[22] や RSL[23] [24] などといった手法の採用が必要と考える。それ故、対策の対象となる SBOX などの暗号演算処理部、具体的には NAND、 NOR セル使用部が小さいほど低面積、低消費電力の耐リークモジュールを実現できると考える。文献 [2] ではリーク解析を実装した各種暗号方式の電力解析の評価結果が記載されており、対策効果を確認したと結論づけている。ただし、対策セルを使用して実装した場合においても文献 [18] までを想定すると、論理的には同じでも実装した際の配線などの容量に依存してマスクの値が区別できると指摘している文献 [26] もあることから、レイアウトにおける対策も必要となることが想定される。これは面積だけではなく、設計工数にも大きく影響することを意味している。対策箇所が少ないほど設計工数の面からも優秀であり、これらは一般的に AES よりも軽量暗号のほうが優位に働くものと思われる。

最新リーク対策手法である EM attack sensor について暗号モジュールを搭載したチップに適用させた場合、電磁波攻撃をするためにはセンサを回避しなければならず、十分な起電力が得られない状況に陥ると推測される。この条件下において SBOX 単体への電磁波解析による鍵抽出を考えた場合、テーブルルックアップ方式で実装された 8 ビットの AES SBOX と多くの軽量暗号で採用されている 4 ビットの SBOX では、SBOX の回路規模に起因する消費電力の少なさから軽量暗号への攻撃のほうが困難になることが推測される。SBOX が小型化になることで、SBOX 以外の回路から発生されるノイズの比率が高くなる以外に、電磁波攻撃するためのコイルの最適なポジジョンの選定も SBOX の消費電力の少なさから見つけにくくなることが想定される。

3.2.2.6　リーク解析の現状調査に関する今後の課題

リーク解析への耐性の優劣に関する暗号アルゴリズム間の比較は文献調査だけでは限界があると考える。対策回路を実装した各種暗号方式に対し、文献 [18] 相当のリーク解析の実施が今後の課題である。

3.2.3　軽量暗号アルゴリズムにおける故障利用攻撃の耐性調査

3.2.3.1　故障利用攻撃の調査概要

文献 [27] をはじめとした、故障注入による鍵の抽出攻撃 DFA(Differential Fault Analysis) の容易性は暗号方式に依存する。DFA の攻撃に関する論文の多くが効率的な攻撃手法をシミュレーションなどを用い理論的に研究しているものであり、例えば実際にレーザを注入して特定段の一つ、あるいは複数の SBOX 等を攻撃して Differential Fault Analysis が可能かどうか評価した論文は皆無である。但し、レーザ装置とステージ装置の自動スキャンにより AES 暗号などを対象として DFA ができるツールは市販されており [28]、特に 1 か所への攻撃を想定しているものについて故障対策なく実装された場合は再現可能と考える。本ツールは対象暗号方式以外の他の暗号方式への応用も可能なものと

思われる。以降、各ブロック暗号に関して理論的な DFA 攻撃の研究事例を挙げる。

攻撃を受けることで想定される故障の種類として、演算器の出力などが一時的に誤り、その値を取り込んでしまうことで故障が発生するテンポラリなものと、中間値を格納するフリップフロップが反転するなど恒久的に値が変わってしまうパーマネントなものが考えられるが、ここでは両方とも実チップにおいて攻撃可能と判断する。前者は演算器の入力となる格納された値には故障が含まれていないことになる。

3.2.3.2 AES への故障利用攻撃の文献調査

鍵長 128 ビット使用時の AES への DFA について文献 [29] によると 8 段目の拡大鍵がストアされた領域への 1 ブロックに対してのフォルト注入攻撃において、1 ペアの結果で 2^8 の空間まで絞り込みが可能とのこと。鍵長 192 ビット使用時ならびに 256 ビット使用時の AES への DFA については文献 [30] によるとそれぞれ 3 ペア、4 ペアの結果で 2^{32} の空間まで絞り込むことが可能とのこと。

3.2.3.3 CLEFIA への故障利用攻撃の文献調査

鍵長 128 ビット使用時の CLEFIA への DFA について文献 [31] によると 2 か所への攻撃、2 ペアで平均 $2^{19.02}$ の探索空間まで絞り込むことが可能としている。文献 [32] によると、CLEFIA への DFA について、鍵長 128 ビット使用時は 2 ペアの攻撃結果のみ、鍵長 192 ビットならびに鍵長 256 ビット使用時では 2 ペアの攻撃結果で平均 $2^{10.78}$ の探索空間まで絞り込むことが可能としている。

鍵長 192 ビットならびに鍵長 256 ビット使用時の CLEFIA への DFA について文献 [33] によると、いずれも 8 ペアの攻撃結果で鍵が判明するとしている。

3.2.3.4 TDES への故障利用攻撃の文献調査

TDES ではないが、Single DES への DFA について文献 [34] によると、特定された single ビットへの攻撃を 12 段目で実施していき 7 ペアを入手すると、ランダムな場所への single ビットの故障注入の場合は 9 ペアを入手すると、それぞれ 99% 以上の確率で 16 段目の鍵が回復できるとしている。

3.2.3.5 PRESENT への故障利用攻撃の文献調査

PRESENT-80/128 への DFA に関して文献 [35] によると、2 バイトのランダムフォルトを 28 段目に注入することで、PRESENT-80 であれば 2 ペア、PRESENT-128 であれば 3 ペアで鍵を回復できるとしている。

3.2.3.6 LED への故障利用攻撃の文献調査ならびに対策に関する特記事項

64 ビットブロック暗号、64 ビット鍵である LED-64 への DFA に関して文献 [36] によると、29 段目に対して故障を注入することで、1 ペアで鍵探索空間を平均で $2^{4.03}$ まで絞り込むことができるとしている (鍵探索空間の調査においてはランダムに生成した誤り暗号文ペア 50 組に対して、実際に攻撃を適用後の鍵候補数から算出)。

LED-64 は拡大鍵として使用する 64 ビットの鍵を全て同じ鍵としており、LED-128 は 64 ビット長 2 組の拡大鍵を交互に使用するため、演算中での拡大鍵の演算は不要である。故に、故障攻撃可能な範囲が狭くなる、対策回路を実装したときの負担低減などのメリットが考えられる (但し、文献 [36] は鍵スケジュール部ではなく、暗号処理中の中間値への攻撃)。

3.2.3.7 Piccolo、 TWINE への故障利用攻撃

研究が開始されたところである。暗号演算部分の 1 ビットあるいは 1 ニブルのレーザ攻撃ではないが、ソフトウェアによる暗号実装において命令への故障攻撃を想定したものとしては、64 ビットブロック暗号で 80 ビット鍵の Piccolo-80、 同じく 64 ビットブロック暗号で 80 ビット鍵の TWINE-80 に対して、正しい暗号文と故障注入により誤った二つの暗号文の組で鍵を抽出できるとしており、128 ビット鍵の CLEFIA-128 より容易という報告は出ている [37]。

3.2.3.8 PRINCE への故障利用攻撃ならびに対策に関する特記事項

64 ビットブロック暗号、128 ビット鍵である PRINCE の 10 段目に対して 1 ニブルの攻撃を実施。1000 例による PC での探索空間調査の結果、4 回の故障注入で 2^{18} 未満の探索空間まで絞り込むことができるとしている [38]。

PRINCE は拡大鍵として使用する 64 ビットの鍵を全て同じ鍵としており、拡大鍵の演算は実装不要である。故に、故障攻撃可能な範囲が狭くなる、対策回路を実装したときの負担低減などのメリットが考えられる (但し、文献 [38] は鍵スケジュール部ではなく、暗号処理中の中間値への攻撃)。

3.2.3.9 複数の暗号方式を対象とした故障利用攻撃の文献調査

多数の暗号方式への故障利用攻撃の最近の調査として文献 [39] が挙げられる。本文献では一般型 Feistel 構造への故障利用攻撃を比較しており、対象は DES(single)、 TWINE、CLEFIA 等。解析のしかたはオーソドックスで、ラウンドの前後でフォルトが伝搬するブロック関係を行列で表記。各段において single ビットの故障を与えたとき、Subkey ブロックのうちアタックされた個数、故障利用攻撃の際に中間値を推測した候補の数をまとめており、過去に発表された論文、例えば文献 [31] との比較を行いながら、本解析手法における故障利用攻撃の最適な攻撃の段数をまとめている。

3.2.3.10 故障利用攻撃耐性のまとめ

鍵長 128 ビット使用時の AES、 LED-64 が 1 ペアで鍵探索空間を 2^8 以下まで絞りこみ可能となっており、耐性が比較的低い。一方、TDES は鍵を 56 ビット毎 3 回に分けて使用するため、故障利用攻撃の耐性が比較的高いと考える。

また、今回調査した軽量暗号への故障利用攻撃の多くが 2012 年から 2014 年に発表されたものであるため、今後の研究により更なる故障注入回数の低減の可能性があると考える。

実チップへの攻撃については、拡大鍵演算部がないなど演算回路規模の小さいもののほうが攻撃範囲が狭いなどの可能性がある。更に、上記にも記載した文献 [29] の AES の攻撃に関しては拡大鍵の演算結果のブロックの一つにパーマネントのエラーを注入することで、中間値だけではなく拡大鍵計算時に故障が伝搬することも利用しており、特にオンザフライによる実装の脆弱性を確認しているが、上記軽量暗号の中には拡大鍵演算実行不要のものが提案されており、拡大鍵の故障伝搬による攻撃が利用できないという点で従来より故障耐性が高いと考えることが出来る。

次に、二回演算ならびに逆算による対策、冗長回路の実装、各種センサの実装による 3 つの主な対策手法において、それぞれ軽量暗号に適用した際の AES と比較しての優劣を記述する。

■二回演算ならびに逆算による対策 対策方法のひとつに文献 [40] に記載されている二回計算 (Doubling)、逆算などが考えられる。文献 [40] では DES を例にとっているが、他の共通鍵暗号方式においても適用可能と考える。但し、文献 [40] ではレーザによる故障利用攻撃において同じ場所に複数回照射する攻撃例や、複数の箇所にレーザを照射させる攻撃でこれらの対策を無効にすることが出来るとしており、攻撃者の能力や攻撃費用を想定して必要相当の対策を講

じることを DES と同様、他の暗号を実装した場合にも求められる。Doubling 対策を実装した実チップへのレーザ攻撃成功例については文献 [41] が挙げられる。複数回演算による対策を施す場合、一般的に暗号処理時間が高速である軽量暗号のほうが AES と比較して追加対策によるレイテンシ増加を抑えることが期待できる。

■冗長回路の実装による対策　冗長化、二重化など追加回路の実装による故障対策も考えられる。例えば文献 [42] で、冗長の程度と検出率を比較している。上記テンポラリーエラーが発生することで故障が注入された場合、単に中間値が格納されているフリップフロップ等に冗長ビットを持たせただけでは検出できない可能性がある。以上から冗長化等による対策の場合、演算器等を含めた暗号実装本体の面積に比例するものと思われ、一般的に軽量暗号が AES などと比較して面積コストの観点から優位に働くものと想定される。なお、上記二回演算等と同様に、冗長回路あるいは多重化された回路と元の回路の双方に攻撃される可能性についての脅威分析は必要であり、分析結果に応じて両方の回路が攻撃された場合の追加の対策が必要となるが、分析の必要性、対策実施の有無は暗号方式には依存しないものと思われる。

■各種センサによる対策　各種故障攻撃を各センサで対処する方法も考えられる。例えば、レーザなどの光源をチップ表面、あるいは裏面から局所的に照射することで故障利用攻撃を試みる手法に対して、光センサをチップ内にちりばめるように実装することで故障を防ぐ方法も提案されている [43]。本方式による対策の場合、光センサ実装による面積増は暗号実装本体の面積に比例するものと想定できることから、一般的に軽量暗号が AES などと比較して面積コストの観点から優位に働くものと想定される。電磁波注入によるチップへの局所攻撃 [44] も出てきているが、暗号アルゴリズムへの故障利用攻撃に使用された場合のセンサ複数配置による対策についても光センサと同様、面積コストの観点から一般的に軽量暗号が優位と考える。電源グリッチ [28] による故障利用解析をセンサ等で防御する場合は、チップ全体の電源回りの設計に大きく依存することになるため、暗号方式による面積コストの優位不利は少ないものと思われる。

3.2.3.11　故障利用攻撃手法の応用

故障利用攻撃の応用として、AES 演算における鍵長 128 ビット使用時の攻撃において Differential ではなく、攻撃により誤った暗号文のみを集めて鍵を復元する試みも文献 [45] でおこなわれている。故障注入の成功率が 50% から 100% それぞれにおいて、ラウンド 7 への攻撃において 4 から 10 の誤ったメッセージで 2^0 から $2^{39.7}$ の鍵候補の絞り込みが 62 から 100% の確率で出来ると調査されている。

3.2.3.12　故障利用攻撃の現状調査に関する今後の課題

故障利用攻撃に関しても文献調査のみならず、厳密には実チップによる各暗号アルゴリズムでの比較対象が望ましい。実チップによる故障利用攻撃、耐性評価は今後の課題である。

参考文献

[1] CRYPTREC Report 2012 暗号実装委員会報告

[2] CCRA. Application of Attack Potential to Smartcards CCDB-2013-05-002, `http://www.commoncriteriaportal.org/cc/`

[3] Pascal Manet, and Bruno Robisson. Differential Behavioral Analysis. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems — CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 413–426. Springer-Verlag, Berlin, Heidelberg, New York, 2007.

[4] Andrey Bogdanov, Lars Ramkilde Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and Charlotte H. Vikkelsoe. PRESENT: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems — CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer-Verlag, Berlin, Heidelberg, New York, 2007.

[5] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer-Verlag, Berlin, Heidelberg, New York, 2011.

[6] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An ultra-lightweight blockcipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 342–357. Springer-Verlag, Berlin, Heidelberg, New York, 2011.

[7] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE: A lightweight, versatile block cipher. In Gregor Leander and François-Xavier Standaert, editors, *ECRYPT Workshop on Lightweight Cryptography*, pages 146–169. ECRYPT II, 2011.

[8] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, Tolga Yalçn, “PRINCE – A Low-Latency Block Cipher for Pervasive Computing Applications”, In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.

[9] Paul Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In Neal Koblitz , editor, *Advances in Cryptology - CRYPTO 1996 - 16th Annual International Cryptology Conference Santa Barbara, California, USA, August 18-22, 1996. Proceedings*, volume 1109 of *Lecture Notes*

in Computer Science, pages 104–113. Springer-Verlag, Berlin, Heidelberg, New York, 1996. `http://www.cryptography.com/public/pdf/TimingAttacks.pdf`

[10] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. Power Analysis Attacks Revealing the Secrets of Smart Cards. 2007 Springer.

[11] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. `http://www.cryptography.com/public/pdf/DPA.pdf`

[12] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In Marc Joye, and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems — CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer-Verlag, Berlin, Heidelberg, New York, 2004.

[13] Thomas S. Messerges. Using Second-Order Power Analysis to Attack DPA Resistant Software. In Çetin K. Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2000*, volume 1965 of *Lecture Notes in Computer Science*, pages 238–251. Springer-Verlag, Berlin, Heidelberg, New York, 2000.

[14] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual Information Analysis. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems — CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer-Verlag, Berlin, Heidelberg, New York, 2008.

[15] S. Chari, J.R. Rao, and P. Rohatgi. Template Attacks. In Burton S. Kaliski, Çetin K. Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 13–28. Springer-Verlag, Berlin, Heidelberg, New York, 2002.

[16] Stefan Mangard, Norbert Pramstaller, and Elisabeth Oswald. Successfully Attacking Masked AES Hardware Implementations. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 157–171. Springer-Verlag, Berlin, Heidelberg, New York, 2005.

[17] D. Agrawal, B. Archambeault, J.R. Rao, and P. Rohatgi. The EM Side-channel(s). In Burton S. Kaliski, Çetin K. Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 29–45. Springer-Verlag, Berlin, Heidelberg, New York, 2002

[18] Takeshi Sugawara, Daisuke Suzuki, Minoru Saeki, Mitsuru Shiozaki, and Takeshi Fujino. On Measurable Side-Channel Leaks inside ASIC Design Primitives. In Guido Bertoni, and Jean-Sébastien Coron, editors, *Cryptographic Hardware and Embedded Systems — CHES 2013*, volume 8086 of *Lecture Notes in Computer Science*, pages 159–178. Springer-Verlag, Berlin, Heidelberg, New York, 2013

[19] `http://www.risec.aist.go.jp/project/sasebo/`

[20] Amir Moradi and Axel Poschmann. Lightweight Cryptography and DPA Countermeasures: A Survey. `http://emsec.rub.de/media/crypto/veroeffentlichungen/2010/09/05/wlc.pdf`

[21] Naofumi Homma, Yu-ichi Hayashi, Noriyuki Miura, Daisuke Fujimoto, Daichi Tanaka, Makoto Nagata, and Takafumi Aoki. EM Attack Is Non-invasive? - Design Methodology and Validity Verification of EM Attack Sensor. In Lejla Batina, and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems — CHES 2014*, volume 8731 of *Lecture Notes in Computer Science*, pages 1–16. Springer-Verlag, Berlin,

Heidelberg, New York, 2014

[22] T. Popp and S. Mangard. Masked Dual-Rail Pre-charge Logic: DPA-Resistance without Routing Constraints. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 172–186. Springer-Verlag, Berlin, Heidelberg, New York, 2005.

[23] D. Suzuki, M. Saeki, and T. Ichikawa. Random Switching Logic: A Countermeasure against DPA based on Transition Probability. Cryptology ePrint Archive, Report 2004/346, 2004. `http://eprint.iacr.org/`

[24] D. Suzuki, M. Saeki, and T. Ichikawa. Random Switching Logic: A New Countermeasure against DPA and Second-Order DPA at the Logic Level. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E90-A(1): pages 160–168. IEICE, 2007.

[25] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. T. M. Shalmani. On the Power of Power Analysis in the Real World: A Complete Break of the KeeLoqCode Hopping Scheme. In Wagner David, editors *Advances in Cryptology - CRYPTO 2008 - 28th Annual International Cryptology Conference Santa Barbara, CA, USA, August 17-21, 2008. Proceedings*, volume 5157 of *Lecture Notes in Computer Science*, pages 203–220. Springer-Verlag, Berlin, Heidelberg, New York, 2008.

[26] Patrick Schaumont and Kris Tiri. Masking and Dual-Rail Logic Don't Add Up. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems — CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 95–106. Springer-Verlag, Berlin, Heidelberg, New York, 2007.

[27] D. Boneh, R. A. DeMillo, and R. J. Lipton. A New Breed of Crypto Attack on "Tamperproof" Tokens Cracks Even the Strongest RSA Code. 1996.

[28] RISCURE 社. `https://www.riscure.com/`

[29] Sk Subidh Ali and Debdeep Mukhopadhyay. A Differential Fault Analysis on AES Key Schedule using Single Fault. In Luca Breveglieri, Sylvain Guilley, Israel Koren, David Naccache, and Junko Takahashi, editors, *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on Date 28-28 Sept. 2011 IEEE computer society*, pages 54–64, IEEE, 2011.

[30] 高橋 順子、福永 利徳Differential Fault Analysis on AES with 192 and 256-bit keys. *2010*年 暗号と情報セキュリティシンポジウム — *SCIS 2010*. 2010.

[31] Junko Takahashi, and Toshinori Fukunaga. Improved Differential Fault Analysis on CLEFIA. In Luca Breveglieri, Shay Gueron, Israel Koren, David Naccache, and Jean-Pieere Seifert, editors, *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2008 Workshop on Date 10-10 Aug. 2008 IEEE computer society*, pages 25–34. IEEE, 2008.

[32] Junko Takahashi, Toshinori Fukunaga. Differential Fault Analysis on CLEFIA with 128, 192, and 256-Bit Keys. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E93-A No.1 pages 136–143. IEICE, 2010

[33] S Ali, and D Mukhopadhyay. Improved Differential Fault Analysis of CLEFIA. In Wieland Fischer, and Jorn-Marc Schmidt, editors, *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2013 Workshop on Date 20-20 Aug. 2013 IEEE computer society*, pages 60–72. IEEE, 2013.

[34] Matthieu Rivain, Emmanuel Prouff, Julien Doget. Differential Fault Analysis on DES Middle Rounds. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems — CHES 2009*, volume 5747 of *Lecture Notes in Computer Science*, pages 457–470. Springer-Verlag, Berlin, Heidelberg, New

York, 2009.

[35] Kitae Jeonga, Yuseop Leea, Jaechul Sungb, and Seokhie Honga. Improved differential fault analysis on PRESENT-80/128. *International Journal of Computer Mathematics, Volume 90, Issue 12*, pages 2553-2563. 2013.

[36] 上野 嶺、 本間 尚文、 青木 孝文. LED 暗号への単一の故障注入を用いた差分故障解析とその評価. *2014* 年 暗号と情報セキュリティシンポジウム — *SCIS 2014*. 2014.

[37] Hideki YOSHIKAWA, Masahiro KAMINAGA, Arimitsu SHIKODA, and Toshinori SUZUKI. Round Addition DFA on 80-bit Piccolo and TWINE. *IEICE Transactions on Information and Systems*, Vol.E96-D No.9 pages 2031–2035. IEICE, 2013.

[38] Ling Song, Lei Hu. Differential Fault Attack on the PRINCE Block Cipher. `http://eprint.iacr.org/2013/043.pdf`

[39] Helene Le Bouder, Gael Thomas, Yanis Linge and Assia Tria. On Fault Injections in Generalized Feistel Networks. *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2014 Workshop on Date 23-23 Sept. 2014 IEEE computer society*, pages 83–93. IEEE, 2014.

[40] Rob Bekkers and Hans König "Fault Injection, a Fast Moving Target in Evaluations", FDTC2011, IEEE computer society, p.65, IEEE. `http://conferenze.dei.polimi.it/FDTC11/shared/FDTC-2011-keynote-2.pdf`

[41] 大野 仁、土屋 遊、 中田 量子、松本 勉. IC カードへのレーザー照射フォールト攻撃は単純な冗長実装では防げない. *2014* 年 暗号と情報セキュリティシンポジウム — *SCIS 2014*. 2014.

[42] Tal G. Malkin, François-Xavier Standaert, Moti Yung. A Comparative Cost/Security Analysis of Fault Attack Countermeasures. In Luca Breveglieri, Israel Koren, David Naccache, and Jean-Pierrei Seifert, editors, *Fault Diagnosis and Tolerance in Cryptography: Third International Workshop, FDTC 2006*, volume 4236 of *Lecture Notes in Computer Science*, pages 159–172. Springer-Verlag, Berlin, Heidelberg, New York, 2006.

[43] Odile Derouet. Secure Smartcard Design against Laser Fault Injection Attacks (invited), FDTC2007, `http://conferenze.dei.polimi.it/FDTC07/Derouet_remaster.pdf`

[44] F. Poucheret, K. Tobich, M. Lisarty, L. Chusseauz, B. Robissonx, and P. Maurine. Local and Direct EM Injection of Power Into CMOS Integrated Circuits. In Luca Breveglieri, Sylvain Guilley, Israel Koren, David Naccache, and Junko Takahashi, editors, *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on Date 28-28 Sept. 2011 IEEE computer society*, pages 100–104, IEEE, 2011.

[45] Thomas Fuhr, Eliane Jaulmes, Victor Lomné and Adrian Thillard. Fault Attacks on AES with Faulty Ciphertexts Only. In Wieland Fischer, and Jorn-Marc Schmidt, editors, *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2013 Workshop on Date 20-20 Aug. 2013 IEEE computer society*, pages 108–118. IEEE, 2013.

3.3 CAESAR プロジェクト

本章では、暗号技術調査 WG (軽量暗号 WG) の外部動向調査として、CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) プロジェクトについてまとめる。本プロジェクトのウェブサイトは `http://competitions.cr.yp.to/caesar.html` である。

3.3.1 CAESAR プロジェクト

■プロジェクト発足の背景　認証暗号は、データの暗号化と認証を同時に行うための共通鍵暗号技術である。AES-CCM や AES-GCM など、すでに標準化され実用化されている認証暗号では、オンラインではない (あらかじめ入力データ長を決めないと処理を開始できない)、計算効率を改善させる余地がある、証明可能安全性に不備がある、弱鍵が存在する、といった様々な問題点が指摘されている。

また、OpenSSH や TLS、802.11 ネットワークにおける WEP などで安全性の問題点が指摘されており、認証暗号はこれらの問題の解決策として期待されている。一方、これらにおいて認証暗号の普及は遅れており、現状の認証暗号の計算効率が、たとえば RC4 などより劣る点にその原因の一つがあると考えられる。

■プロジェクトの目標　本プロジェクトの目標は、(1) AES-GCM より (安全性、計算効率、実装効率、あるいはその他何らかの点において) 優れていて、なおかつ (2) 広範に実用化されることに適した認証暗号のポートフォリオを選定することにある。

■プロジェクトの概要　本プロジェクトでは認証暗号アルゴリズムの公募を行う。応募締め切りは 2014 年 3 月であり、誰でも応募が可能である。公募されたアルゴリズムは第一ラウンドアルゴリズムであり、おおよそ 1 年の評価期間を経て 2015 年 1 月に第二ラウンド進出アルゴリズムを決定する。その後さらに 1 年の評価期間を経て 2015 年 12 月に第三ラウンド進出アルゴリズムを決定し、さらに 1 年の評価期間を経て 2016 年 12 月に最終候補アルゴリズムを決定する。ポートフォリオのアナウンスは 2017 年 12 月を予定している。

本プロジェクトは研究者主体で進められるものであり、ポートフォリオは標準を意味するものではない (ただし、本プロジェクトは NIST によるスポンサーシップを受けている)。また、各ラウンドに進出するアルゴリズムの決定では、選定委員による投票が行われる予定である。

本プロジェクトでは各提案者の設計指針に応じて安全性、機能、実装性能、計算効率など様々な評価要素が考えられ、「軽量」性についても評価要素に入ることが予想される。

■スケジュール詳細　下記スケジュールを予定している[*2]。

- M-20, 2012.07.05–06: DIAC: Directions in Authenticated Ciphers. Stockholm.
- M-14, 2013.01.15: Competition announced at the Early Symmetric Crypto workshop in Mondorf-les-Bains; also announced online.
- M-7, 2013.08.11–13: DIAC 2013: Directions in Authenticated Ciphers 2013. Chicago.
- M0, 2014.03.15: Deadline for first-round submissions.
- M1, 2014.05.15: Deadline for first-round software.

[*2] 2015 年 2 月 20 日現在。頻繁に更新されており、最新情報はプロジェクトのウェブサイト `http://competitions.cr.yp.to/caesar.html` より確認できる。

- M5 2014.08.23–24: DIAC 2014: Directions in Authenticated Ciphers 2014. Santa Barbara.
- M12 (tentative), 2015.03.15: Announcement of second-round candidates.
- M13 (tentative), 2015.04.15: Deadline for second-round tweaks.
- M14 (tentative), 2015.05.15: Deadline for second-round software.
- M15 (tentative), 2015.06.15: Deadline for second-round Verilog/VHDL.
- 2015 summer (tentative): DIAC 2015.
- M21 (tentative), 2015.12.15: Announcement of third-round candidates.
- M22 (tentative), 2016.01.15: Deadline for third-round tweaks.
- M23 (tentative), 2016.02.15: Deadline for third-round software.
- M24 (tentative), 2016.03.15: Deadline for third-round Verilog/VHDL.
- 2016 summer (tentative): DIAC 2016.
- M33 (tentative), 2016.12.15: Announcement of finalists.
- M34 (tentative), 2017.01.15: Deadline for finalist tweaks.
- M35 (tentative), 2017.02.15: Deadline for finalist software.
- M36 (tentative), 2017.03.15: Deadline for finalist Verilog/VHDL.
- 2017 summer (tentative): DIAC 2017.
- M45 (tentative), 2017.12.15: Announcement of final portfolio.

■公募要領　2014 年 1 月 27 日に公募要領の最終版が公表された。2014 年 3 月の応募時点では下記情報を含めたドキュメントを提出する。

- 方式の名称、設計者、応募者、連絡用メールアドレス
- 仕様
- 安全性のゴール
- 安全性解析
- 特筆すべき事項、特徴
- 設計の合理性
- 知的財産に関する事項
- 応募に際して合意する事項

その後 2014 年 5 月 15 日までにソフトウェアでのレファレンスコードを提出する。また、第二ラウンド進出アルゴリズムについては、応募者は 2015 年 4 月までにハードウェアでのレファレンス実装を提出する。

■選定委員　選定委員は下記 22 名のメンバーからなる。

1. Steve Babbage (Vodafone Group, UK)
2. Daniel J. Bernstein (University of Illinois at Chicago, USA, and Technische Universiteit Eindhoven, Netherlands); secretary, non-voting
3. Alex Biryukov (University of Luxembourg, Luxembourg)
4. Anne Canteaut (Inria Paris-Rocquencourt, France)
5. Carlos Cid (Royal Holloway, University of London, UK)
6. Joan Daemen (STMicroelectronics, Belgium)

7. Christophe De Cannière (Google, Switzerland)
8. Orr Dunkelman (University of Haifa, Israel)
9. Henri Gilbert (ANSSI, France)
10. Tetsu Iwata (Nagoya University, Japan)
11. Lars R. Knudsen (Technical University of Denmark, Denmark)
12. Stefan Lucks (Bauhaus-Universität Weimar, Germany)
13. David McGrew (Cisco Systems, USA)
14. Willi Meier (FHNW, Switzerland)
15. Kaisa Nyberg (Aalto University School of Science, Finland)
16. Bart Preneel (COSIC, KU Leuven, Belgium)
17. Vincent Rijmen (KU Leuven, Belgium)
18. Matt Robshaw (Impinj, USA)
19. Phillip Rogaway (University of California at Davis, USA)
20. Greg Rose (Qualcomm Technologies Inc., USA)
21. Serge Vaudenay (EPFL, Switzerland)
22. Hongjun Wu (Nanyang Technological University, Singapore)

■応募方式一覧　下記の 57 方式が提案された。方式の名称と設計者を記載している。冒頭の (L) は、軽量性を特徴として挙げている方式を示している[*3]。

1. (L) ACORN: v1 (Hongjun Wu)
2. (L) ++AE: v1.0 (Francisco Recacha)
3. AEGIS: v1 (Hongjun Wu, Bart Preneel)
4. AES-CMCC: v1, v1.1 (Jonathan Trostle)
5. AES-COBRA: v1, withdrawn, (Elena Andreeva, Andrey Bogdanov, Martin M. Lauridsen, Atul Luykx, Bart Mennink, Elmar Tischhauser, Kan Yasuda)
6. AES-COPA: v1 (Elena Andreeva, Andrey Bogdanov, Atul Luykx, Bart Mennink, Elmar Tischhauser, Kan Yasuda)
7. AES-CPFB: v1 (Miguel Montes, Daniel Penazzi)
8. (L) AES-JAMBU: v1 (Hongjun Wu, Tao Huang)
9. AES-OTR: v1 (Kazuhiko Minematsu)
10. AEZ: v1 (Viet Tung Hoang, Ted Krovetz, Phillip Rogaway)
11. Artemia: v1 (Javad Alizadeh, Mohammad Reza Aref, Nasour Bagheri)
12. (L) Ascon: v1 (Christoph Dobraunig, Maria Eichlseder, Florian Mendel, Martin Schläffer)
13. AVALANCHE: v1 (Basel Alomair)
14. (L) Calico: v8, withdrawn, (Christopher Taylor)
15. CBA: v1 v1-1 (Hossein Hosseini, Shahram Khazaei)
16. (L) CBEAM: r1, withdrawn, (Markku-Juhani O. Saarinen)

[*3] “lightweight” をキーワードとして応募ドキュメントを検索し、軽量性を方式の特徴として挙げているか、あるいは使用している演算や構成要素を軽量性を考慮して選定している方式をピックアップした。

17. CLOC: v1 (Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka)
18. (L) Deoxys: v1 (Jérémy Jean, Ivica Nikolić, Thomas Peyrin)
19. (L) ELmD: v1 (Nilanjan Datta, Mridul Nandi)
20. Enchilada: v1 v1.1 (Sandy Harris)
21. (L) FASER: v1, withdrawn, (Faith Chaza, Cameron McDonald, Roberto Avanzi)
22. HKC: v1, withdrawn, (Matt Henricksen, Shinsaku Kiyomoto, Jiqiang Lu)
23. HS1-SIV: v1 (Ted Krovetz)
24. ICEPOLE: v1 (PawełMorawiecki, Kris Gaj, Ekawat Homsirikamol, Krystian Matusiewicz, Josef Pieprzyk, Marcin Rogawski, Marian Srebrny, Marcin Wojcik)
25. iFeed[AES]: v1 (Liting Zhang, Wenling Wu, Han Sui, Peng Wang)
26. (L) Joltik: v1 (Jérémy Jean, Ivica Nikolić, Thomas Peyrin)
27. Julius: v1.0 (Lear Bahack)
28. (L) Ketje: v1 (Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche, Ronny Van Keer)
29. Keyak: v1 (Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche, Ronny Van Keer)
30. (L) KIASU: v1 (Jérémy Jean, Ivica Nikolić, Thomas Peyrin)
31. (L) LAC: v1 (Lei Zhang, Wenling Wu, Yanfeng Wang, Shengbao Wu, Jian Zhang)
32. Marble: v1.0 (Jian Guo)
33. McMambo: v1, withdrawn, (Watson Ladd)
34. (L) Minalpher: v1 (Yu Sasaki, Yosuke Todo, Kazumaro Aoki, Yusuke Naito, Takeshi Sugawara, Yumiko Murakami, Mitsuru Matsui, Shoichi Hirose)
35. MORUS: v1 (Hongjun Wu, Tao Huang)
36. NORX: v1 (Jean-Philippe Aumasson, Philipp Jovanovic, Samuel Neves)
37. OCB: v1 (Ted Krovetz, Phillip Rogaway)
38. OMD: v1.0 (Simon Cogliani, Diana-Ştefania Maimuţ, David Naccache, Rodrigo Portella do Canto, Reza Reyhanitabar, Serge Vaudenay, Damian Vizár)
39. PAEQ: v1 (Alex Biryukov, Dmitry Khovratovich)
40. PAES: v1, withdrawn, (Dingfeng Ye, Peng Wang, Lei Hu, Liping Wang, Yonghong Xie, Siwei Sun, Ping Wang)
41. PANDA: v1, withdrawn, Dingfeng Ye, Peng Wang, Lei Hu, Liping Wang, Yonghong Xie, Siwei Sun, Ping Wang)
42. (L) π-Cipher: v1 (Danilo Gligoroski, Hristina Mihajloska, Simona Samardjiska, Håkon Jacobsen, Mohamed El-Hadedy, Rune Erlend Jensen)
43. POET: v1 (Farzaneh Abed, Scott Fluhrer, John Foley, Christian Forler, Eik List, Stefan Lucks, David McGrew, Jakob Wenzel)
44. POLAWIS: v1 (Arkadiusz Wysokinski, Ireneusz Sikora)
45. (L) PRIMATEs: v1 (Elena Andreeva, Begül Bilgin, Andrey Bogdanov, Atul Luykx, Florian Mendel, Bart Mennink, Nicky Mouha, Qingju Wang, Kan Yasuda)
46. (L) Prøst: v1 (Elif Bilge Kavun, Martin M. Lauridsen, Gregor Leander, Christian Rechberger, Peter Schwabe, Tolga Yalçın)
47. Raviyoyla: v1 (Rade Vuckovac)

48. (L) Sablier: v1 (Bin Zhang, Zhenqing Shi, Chao Xu, Yuan Yao, Zhenqi Li)
49. (L) SCREAM: v1 (Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, Kerem Varici, François Durvaux, Lubos Gaspar, Stéphanie Kerckhof)
50. SHELL: v1 (Lei Wang)
51. (L) SILC: v1 (Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, Sumio Morioka, Eita Kobayashi)
52. Silver: v1 (Daniel Penazzi, Miguel Montes)
53. STRIBOB: v1 (Markku-Juhani O. Saarinen)
54. Tiaoxin: v1.0 (Ivica Nikolić)
55. TriviA-ck: v1 (Avik Chakraborti, Mridul Nandi)
56. Wheesht: v1 (Peter Maxwell)
57. YAES: v1 v2 (Antoon Bosselaers, Fre Vercauteren)

3.3.2 まとめ

本章では、暗号技術調査 WG (軽量暗号 WG) の外部動向調査として、CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) プロジェクトについてまとめた。AES コンペティション、NESSIE プロジェクト、eSTREAM プロジェクト、SHA-3 プロジェクトに続く国際的なコンペティションであり、継続的に注視していくことが求められる。

3.4 軽量暗号の活用事例および標準化動向調査

3.4.1 調査目的

今後の暗号の開発において、活用事例・標準化動向から軽量暗号に関する要求条件を導き出し、研究開発、標準化の指針を得る。

3.4.2 活用事例調査

3.4.2.1 調査方法

以下に示す軽量暗号が活用されると期待されている分野について公開されている情報を調査する。

- RFID
- センサーネットワーク（環境測定等）
- 医療センサ
- ITS、自動車
- 記録メディア（HDD、SSD 等）
- 携帯端末（携帯電話、タブレット端末、ポータブルゲーム機等）
- その他

3.4.2.2 調査報告

3.4.2.1 章に挙げたそれぞれの項目について調査を行った。現段階で実際に軽量暗号が使用されているという公開の情報はない。そこで、これらの項目について、暗号についてどのように利用されているか、その中で軽量暗号がどのような導入方法が考えられるかを考察する。

■RFID、センサーネットワーク（環境測定等）、携帯端末（携帯電話、タブレット端末、ポータブルゲーム機等）　これらについては、一般論となる情報のみが公開されていた [1, 2, 3]。無線ネットワーク接続機能を持つ RFID がインターネットのようなオープンなネットワークに接続する際に暗号を利用する。ほとんどの RFID、センサー、携帯端末デバイスの CPU は低スペックであり、信号処理を行う能力に乏しいこと、またメモリサイズも小さい。さらに、低消費電力での実装をしなければならず、軽量暗号に対する期待は極めて大きい。

■医療センサ　Texas Instrument[4] では、血圧、体温、心拍数、ブトウ糖等の測定を行い、Bluetooth で通信を行い、低消費電力のポータブル機器用のデバイス、MSP430FR59xx ファミリーを提供している。このデバイスの記事では、“政府標準である”256-bit AES を用いた医療用センサと記載されている。MSP430FR59xx ファミリーのスペックでは、MSP430microcotroller（16bitRISC CPU）を使用、メモリサイズは 32KB-256KB、消費電力は不明である。また、Position Paper[5] での報告として遠隔健康ケアシステムでは、埋め込みデバイスが使われており、これらとのコミュニケーションをとる際にセキュリティ技術が必要。そして、これらを Ultra-low-power で行いたいとしている。この論文では、待機電力を減らす、置換を減らすなどによるアルゴリズムの簡素化が主体で低消費電力化を図っている。

■ITS、自動車　ITS（Intelligent Transport Systems:高速道路交通システム）は、人と道路と自動車の間で情報の共有を行い、交通の最適化を図るシステムとして作られたシステムである [6]。そして、”安全”が強調されたシステム造

りが目指されている。そして、その基本構成である自動車搭載機器について、装置機器間のデータの秘匿、認証のために暗号が利用されることが謳われている [7, 8, 9, 10]。これらについても、リソースが限られているとはわかっているものの、軽量暗号を使用する段階には至っておらず、軽量暗号を使うことを提案している段階 [11] である。

■記録メディア（HDD、SSD 等）　SandForce[12] ではセキュリティ機能を持つ SSD を提供している。Windows8 の PC やタブレットにおいても低消費電力の記録デバイス（SSD）が必要であった。このため、従来品が 20mA を利用していたのに対し、0.05mA で動作するようにしている。モバイル端末でも利用可能であるとのこと。

■その他　ICT 社会において、ETC システムにおいても暗号化が使われている。このシステムではプライバシ保護のため、認証、暗号化などが必要となる [13]。　その他、軽量暗号の一般的な利用可能性については、軽量暗号関係の多くの論文 [14, 15] で書かれている。

3.4.2.3　ヒアリング

メーカー数社にヒアリングを行い、軽量暗号に対する考え方を調査した。その結果を以下に示す。最近の測定器や家電はほぼ CPU が積まれ、ネットワーク接続が可能となっているが、使用目的によって要求条件が異なっている。大きくわけて以下の 2 つのケースがある。

1. 一つのハードウェアもしくはソフトウェアに入れる機能が確定している場合
2. 一つのハードウェアもしくはソフトウェアに入れる機能が確定しておらず、いろいろな機能を入れる場合

前者は医療センサのように使用用途が厳密に限られる場合、後者は PC やタブレットのように汎用の機器であり、使用目的が厳密に制限されていない場合である。

■ケース 1 について

- セキュリティが必要で AES を入れたければ、外部モジュールとして AES チップを使う、もしくは、ソフトウェアとして AES を入れられるスペックの CPU やメモリを搭載する。
- 軽量暗号をあえて導入する必要を感じていない。チップサイズ（消費電力を含む）が小さくなればよいという一般論があるが、どれほど小さいチップが必要であるかの指定はない。

■ケース 2 について

- CPU、メモリなどのリソースをそれぞれの機能でシェアして使用する。欠点として、機能が多くなりリソースを取り合うことが生じる。
- ハードウェア、ソフトウェアの構築段階でどの機能を必須として使うかを決める。これにより、リソースを分割する度合い（量、順位）が決まる。
- AES が使用困難であり、軽量暗号であれば使用可能、というアプリケーションは少ない。
- リソースを削られたとしても高速な動作が保障されるような暗号方式として軽量暗号が求められていることはある。

3.4.2.4　活用事例のまとめ

軽量暗号に対する要求はあるものの、具体的なスペックまで落とし込んだ要求条件は出ていない。ただし、医療センサの項で紹介した Texas Instrument のように、標準として認められることで使用する業者があるということも事実で

ある。従って、CRYPTREC などの機関で調査・評価することは、軽量暗号の利用促進に供する情報を提供できると考えられる。

3.4.3 標準化動向調査

3.4.3.1 調査方法

軽量暗号の標準化について、ISO/IEC JTC 1/SC 27/WG 2 で進められてきた標準化内容を調査する。また、IETF Light-Weight Implementation Guidance(lwig) で行われている軽量暗号の実装に関するガイダンスについてまとめる。

3.4.3.2 ISO/IEC JTC 1/SC 27/WG 2 の活動

ISO/IEC 29192 は、チップサイズ、ハードウェアの消費電力、ソフトウェアのコードサイズ、RAM サイズ、伝送容量、実行時間などの制限がある場合の、データ秘匿、認証、本人識別、否認防止、鍵交換などを目的に適した軽量暗号の仕様を標準化している。ISO/IEC JTC 1/SC 27/WG 2 では技術カテゴリに対応し以下の 4 つのパートに分かれてこの標準化作業が行われた。これらの標準化作業はすべて 2013 年までに終了し、各パートの内容が ISO/IEC 29192-1，-2，-3，-4 として標準化されている。

■パート 1：総論　安全性要件、ハードウェア／ソフトウェア実装要件が規定された。

■安全性要件　80 ビットセキュリティ以上

■ハードウェア実装要件　ハードウェアチップ面積、実行サイクル数、1 サイクル当たりの処理ビット数、消費電力、消費電力量、1 ビット当たりの消費電力量、実装に用いられたルールが方式比較のための参考情報とされた。但し、これらの具体的な数値はアプリケーション依存であるため、規定しない。

■ソフトウェア実装要件　プログラムコードサイズ、RAM サイズ、実行速度が方式比較のための参考情報とされた。但し、これらの具体的な数値はアプリケーション依存であるため、規定しない。

■他の特性　　軽量暗号は、短い平文、暗号文に対する処理も重要な要素となる。可能であれば、その特性が示されるべきである。また、実装に伴う遅延についても重要な要素となる。

■パート２：ブロック暗号　２つのブロック暗号、PRESENT と CLEFIA が標準化された。

・PRESENT　ブロックサイズ 64 ビット、鍵サイズ　80、128 ビット
・CLEFIA　　ブロックサイズ 128 ビット、鍵サイズ　128、192、256 ビット

■パート３：ストリーム暗号　２つのストリーム暗号、Enocoro と Trivium が標準化された。

・Enocoro　鍵サイズ 80、128 ビット
・Trivium　鍵サイズ 80 ビット

■パート４：公開鍵暗号（非対称暗号）技術を用いたメカニズム　公開鍵暗号技術を用いた、楕円上の離散対数問題をベースにした認証方式（cryptGPS）と、公開鍵暗号をベースにしたセッション鍵生成・鍵交換方式 (ALIKE) と、Identity ベース署名方式の 3 つが標準化された。

3.4.3.3 IETF における軽量暗号の実装に関するガイダンス

建物、車、電化製品などで使われている多くのデバイスでコミュニケーションができるようになってきた。但し、これらのデバイスの能力は様々であり、能力の小さいデバイスもある。IETF Light-Weight Implementation Guidance (lwig) では、このような小さい能力のデバイスに焦点をあて、非常に制限された環境下で、最小限の IP 接続を可能とする軽量暗号の実装方法、について標準化することを目的とする [16]。現在、lwig で議論が開始された段階であり、インターネット上での鍵交換関連 [17]、モーバイルネットワークでの低消費電力デバイス関連 [18]、TLS のカスタマイズ関連 [19] などの寄与文書が WG に提出されてきているが、まだ RFC 化されたものはない。

なお、lwig とは独立であるが、CLEFIA の暗号アルゴリズムが RFC6114 となっている。

参考文献

[1] PRWeb,“IEC and ISO adopt lower power encryption standard Enocoro stream cipher,”http://www.prweb.com/releases/2012/11/prweb10132688.htm

[2] M. B. Abdelhalim, M.El-Mahallawy, and A. Elhennawy,“Design & Implementation of an Encryption Algorithm for use in RFID System,”International Journal of RFID Security and Cryptography, Vol.1, Issues1-4, Mar-Dec. 2012.

[3] TechRepublic, “Is wireless RFID sensor authentication / encryption possible? Maybe.” http://www.techrepublic.com/blog/it-security/is-wireless-rfid-sensor-authentication-encryption-possible-maybe/

[4] TEXAS INSTRUMENT, “Ultra-low Power Microcontrollers for Portable Medical Device Designs,” http://www.engineering.com/ElectronicsDesign/ElectronicsDesignArticles/ArticleID/6222/ Ultra-low-Power-Microcontrollers-for-Portable-Medical-Device-Designs.aspx

[5] F. H. Qi Hao, and M. Lukowiak, “Implantable Medical Device Communication Security: Pattern vs. Signal Encryption (Position Paper),” https://www.usenix.org/legacy/evnet/healthsec11/tech/final_files/hu-healthsec11.pdf

[6] ITS Japan,「ITS とは」, http://www.its-jp.org/about/

[7] IPA,「2012 年度自動車の情報セキュリティ動向に関する調査」, http://www.ipa.go.jp/files/000027274.pdf

[8] EVITA,“E-safety vehicle intrusion protected applications,” http://www.evita-project.org/

[9] PRESERVE,“PRESERVE preparing secure v2x communication systems,” http://www.evita-project.org/

[10] SAE,“Vehicle Electrical System Security Committee,” http://www.sae.org/works/committeeHome.do?comtID=TEVEES18

[11] 野島，盛合,「『シェア暗号』を自動車に」, http://techon.nikkeibp.co.jp/article/COLUMN/20140401/343501/?rt=nocnt

[12] The TECH REPORT, “SandForce Improves SSD encryption, power management,”http://techreport.com/news/24894/sandforce-improves-ssd-encryption-power-management

[13] 松井,「情報セキュリティ基盤技術暗号技術の最新動向- Cryptography:Technology and Applications -」, http://hiroshi1.hongo.wide.ad.jp/hiroshi/files/toku1/material/Matsui_Mitsubishi.pdf

[14] Axel Poschmann,“Lightweight Cryptography,” http://mathsci.ucd.ie/~gmg/ECC2007Talks/poschmann_LWC.pdf

[15] 鈴木，菅原，佐伯,「軽量／低遅延暗号のハードウェア実装性能について」, SCIS2014, 2A2-2

[16] IETF, “Light-Weight Implementation Guidance (lwig),”https://ietf.org/wg/lwig/charter/

[17] T. Kivinen, "Minimal IKEv2 draft-ietf-iwig-ikev2-minimal-01", https://ietf.org/doc/draft-ietf-lwig-ikev2-minimal/

[18] J. Arkko, A. Eriksson, and A. Keranen, "Minimal IKEv2 draft-ietf-iwig-ikev2-minimal-01," https://ietf.org/doc/draft-ietf-lwig-cellular/

[19] S. S. Kumar, S. Keoh, and H. Tschofenig, "Minimal IKEv2 draft-ietf-iwig-ikev2-minimal-01," https://ietf.org/doc/draft-ietf-lwig-tls-minimal/

第 4 章

軽量暗号のアプリケーションに関するヒアリング

2013 年度第 2 回軽量暗号 WG にて、エンドユーザーからのヒアリングとして、下記の 2 名の方から自動車および社会インフラへの軽量暗号技術の応用について意見を伺った。

- 「自動車における IT セキュリティ」 (トヨタ IT 開発センター 小熊 寿氏)
- 「制御システム向け暗号の要件の考察」 (日立製作所 大和田 徹氏)

小熊氏からは、自動車における IT セキュリティでは、例えば、車載ネットワーク CAN のデータ長が 8 バイトであることから、軽量暗号は、MAC を生成するアルゴリズムとして処理性能や MAC サイズの点で AES よりも有利と思われるとのコメントがあった。

また、大和田氏からは、課題からみた制御システム向け暗号の要件が抽出され、高速処理、低処理負荷、柔軟な暗号化対象長、低リソースでの鍵管理・更新機能等の要件で軽量暗号が役立つ可能性があるとコメントがあった。

2013 年度第 2 回軽量暗号 WG での発表資料を、参考資料として本報告書の A.1 章に掲載している。

第 5 章

軽量ブロック暗号の実装詳細評価

第 2 章で行った現状調査にも軽量暗号の実装評価は含まれるが、既存文献では評価環境や実装者が異なるため、暗号アルゴリズム間の比較が困難であった。そこで、情報通信研究機構にて、軽量ブロック暗号 (AES, Camellia, CLEFIA, PRESENT, LED, Piccolo, TWINE, PRINCE) について、同一プラットフォーム上で、同一の実装者または統一的な実装ポリシーによりハードウェア実装およびソフトウェア実装の評価を行い、統一的な評価環境で比較調査を実施した。この評価結果が 2013 年度 第 3 回軽量暗号 WG にて報告された。実装環境および測定指標は下記の通りである。

■ハードウェア実装評価

- 標準的な CMOS セルライブラリ：NANGATE Open Cell Library (45nm CMOS)
- unrolled 実装, round 実装, serial 実装の 3 通りのアーキテクチャ
- 測定指標：最大動作周波数、処理速度、ゲートカウント、サイクルカウント、消費電力、ピーク電流

■ソフトウェア実装評価

- プロセッサ：ルネサスエレクトロニクス RL78 (16bit 組み込みマイコン)
- 測定指標：処理速度, RAM サイズ, ROM サイズ。ROM, RAM サイズに関して下記 4 通りの組み合わせで、それぞれの範囲内で処理速度を最大化する実装を行った。

ROM	512 B	1024 B
RAM	64 B	128 B

■評価結果概要　ハードウェア実装評価では、軽量暗号は AES と比較して 1-2kgate 回路規模が小さく、この違いはマチュアなプロセス (180nm-350nm) において実装の可否に影響する場合があり、アドバンテージとなること、リアルタイムのメモリ暗号化や μ 秒クラスのリアルタイム通信などのアプリケーションにおいて優位となる可能性があることが報告された。また、小さい、速いという一つの指標だけだと AES との差分が少ないが、小さく、速く、サイドチャネル対策が容易という複数の軸で比較したときに AES に対する優位性がより明確になると報告された。

ソフトウェア（組み込みマイコン）実装においては、コードサイズの小さい暗号への要求が高い。メモリが十分あれば（例えば、アルゴリズム単体で暗号復号込みで ROM 1KB あれば）AES で十分である。よって組み込みマイコンにおいて AES より価値ある軽量ブロック暗号は、暗号・復号込みで ROM 200 B 以下、RAM 32 B 以下でそれなりの速度が達成できるアルゴリズムと考えられるという報告があった。

2013 年度 第 3 回軽量暗号 WG での発表資料を、参考資料として本報告書の A.2 章に掲載している。

付録 A

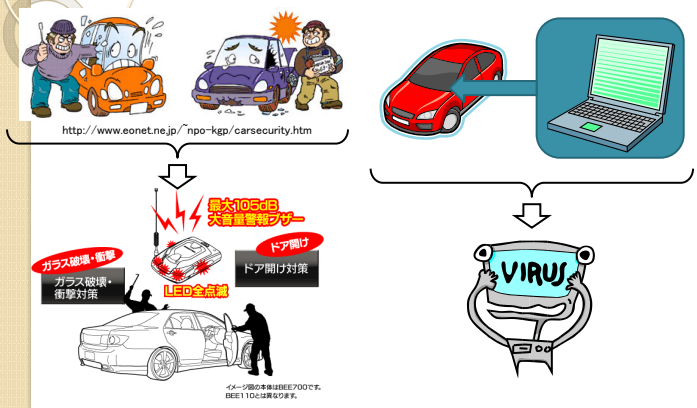
参考資料

A.1　軽量暗号のアプリケーションに関するヒアリング

A.1.1　自動車における IT セキュリティ

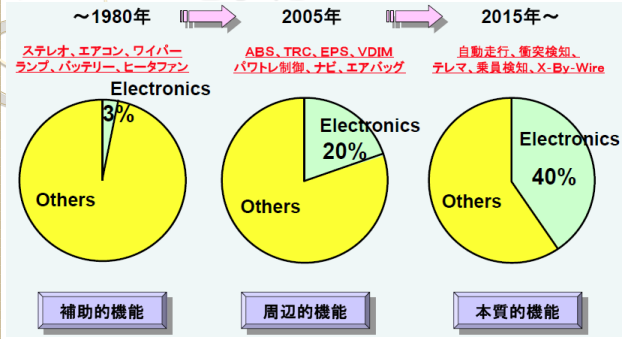
2013 年度 第 2 回軽量暗号 WG (2013 年 12 月 26 日) でのトヨタ IT 開発センター 小熊 寿氏による発表資料を示す。

「クルマ」×「セキュリティ」



http://www.kato-denki.com/products/bee/bee110/img/standard_sensor.png

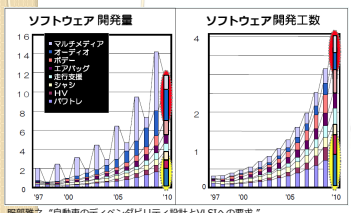
クルマの電子化

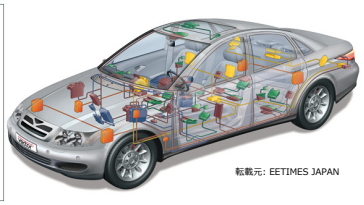


服部雅之, "自動車のディペンダビリティ設計とVLSIへの要求,"
JST CREST「ディペンダブルVLSIシステムの基盤技術」研究領域 平成19年度ワークショップ

- 1970年代の排ガス規制
  - エンジン状態を監視し、燃料噴射量と点火タイミングを制御
  - 電子部品導入のきっかけ

現在のクルマ





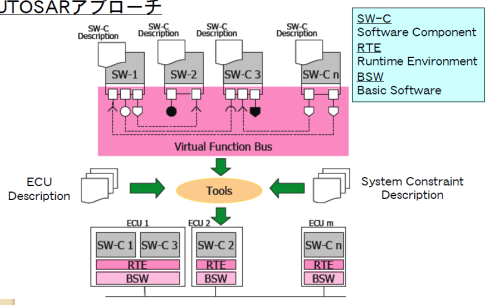
転載元: EETIMES JAPAN

服部雅之, "自動車のディペンダビリティ設計とVLSIへの要求,"
JST CREST「ディペンダブルVLSIシステムの基盤技術」研究領域 平成19年度ワークショップ

- 高級車の事例: LEXUS LS460
  - 100個以上の車載制御マイコン (ECU: Electronic Control Unit)
  - SWの総ライン数はカーナビなどを含めると17M行: 雑誌報道による数値
    - 2015年には100M行という予測も...
- ECUが連携してサービスを提供
  - e.g. 車体姿勢制御、プリクラッシュセーフティ
  - SWの大規模化と複雑化↑

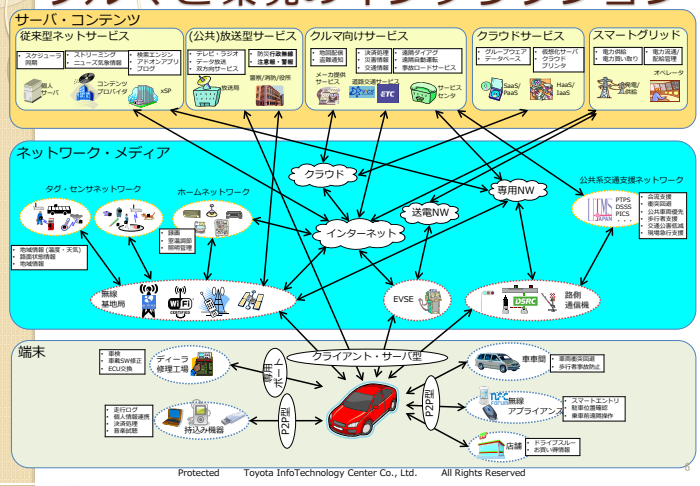
プラットフォームベース開発

AUTOSARアプローチ



- 基盤機能共通化による開発工数およびコストの削減
  - プラットフォームベースの開発へ移行
- AUTOSAR (Automobive Open System Architecture): 2003～
  - SW基盤の業界標準を作成
- JasPar (Japan Automotive Software Platform): 2004～
  - 国内メーカの要望を集約し、AUTOSARへのインプットなどを行う

クルマと環境のインタラクション



標準化とコンピュータ

- 攻撃者がクラック出来る環境
  - 仕様書が閲覧可能であり、セキュリティホールも確認出来る
- 全く同じ事象が起きる環境
  - 1台さえ解析できれば、他も同じアーキテクチャであるため同じ手法が適用可能
- 情報伝搬速度が早い
  - ほとんどのPCがネットワーク接続しているため、瞬時に広範囲に影響する

- アーキテクチャの変化：「機種毎」⇒「みんな同じ」
- 仕様書の公開

- 外部からの侵入口
- 情報発信

1980年代後半
機能の標準化

1990年代前半
インターネット接続

メインフレーム

PC/AT互換機

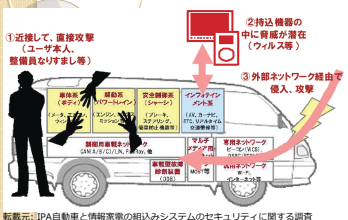
ワークステーション

- 「標準化」と「NW接続」する将来のクルマ
  - 同じサイクルにはまる可能性大

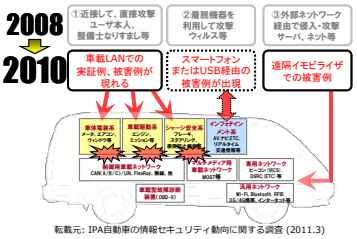
クルマの要求条件

- ハードリアルタイムとFail-Safe
  - 生命に直結するため時間制約が厳しい (即時応答性)
  - 不具合が起きたときに安全側に倒れる事
- 10年以上の耐用年数
  - 製造から廃車までの時間が長く、中古車市場にも転用
- 不具合発生を事前に防止
  - PC: ウイルス感染などの被害が現れてからの対応が多い (セキュリティSWメーカによる事前調査もある)
  - クルマ: 事故など具体的な被害が出る前に対応する必要あり
- 切断時動作
  - NW接続はモバイル機器と同様に無線: 生命に直結するサービスを常時接続前提で考えてはいけない
- 劣悪な環境での動作、信頼性
  - 電圧変動±50% 動作環境温度−40〜140℃

2010年に潮目が変化



転載元: IPA自動車と情報家電の組込みシステムのセキュリティに関する調査 (2009.3)



転載元: IPA自動車の情報セキュリティ動向に関する調査 (2011.3)

- 想定される脅威の列挙
  - 「机上の空論」: 〜のでは?
  - 想定ベースの議論であり、説得性に欠ける: Evidence 不足
- 攻撃結果の公開
  - 予想した脅威が現実化
  - 事例として対外発表が行われる
- 具体的な事故は未報告
  - 想定外の事象が起こる可能性は否定できない

周辺動向

- 海外
  - 国プロ関連
    - 欧州によるFramework Program: 製品化を見据えた研究・標準化活動
      - SeVeCom (FP6): 路車間・車車間通信のセキュリティ
      - EVITA (FP7): 車載システムのセキュリティ
  - 学会
    - Escar: Embedded Security in Cars Conference
  - 産業界
    - SAE Vehicle Electrical System Security Committee (2011〜): NHTSAによるCybersecurity Researchと連携
- 国内
  - 情報処理推進機構による研究会
    - 2006年、2008年〜2011年: クルマ向け情報セキュリティ動向と想定される脅威を調査
  - 産業界
    - 自動車技術会による情報セキュリティ小委員会 (2010〜)

escar: Embedded Security in Cars Conference

- 2003年から毎年ドイツ国内で開催
  - 2003年は20名程度、2008年からはCFP

	2009年	2010年	2011年	2012年	2013年
参加者	54	72	74	112	110
うち日本人	1	3	5	7	16

*参加者リストを参照

- 2013年からはUS、2014年からはアジアでも実施
  - 次回escar USAは7月、CFPの予定
  - 1st escar ASIAは4月

Trusted Assurance Levels

- C2C-CCにて検討中

TAL	Security Evaluation Requirements			Resulting Security Implications		
	Minimum Target of Evaluation (TOE)	Minimum Evaluation Assurance Level (EAL)	Minimum (Hardware) Security Functionality	Prevented (Internal) Attacker acc. to CC	Potential Security Implications	V2X Use Case Examples
0	None	None	None	None	Not reliable against security attacks	Some limited (e.g., using trusted V2I infrastructures)
1	+ V2X software	EAL 3	Only software security mechanisms	Basic	Not reliable against simple hardware attacks (e.g., offline flash manipulation)	Non-safety, but most privacy relevant use cases
2	+ V2X hardware	EAL 4	+ dedicated hardware security (i.e., secure memory & processing) + tamper evidence	Enhanced Basic	Not reliable against more sophisticated hardware attacks (e.g., side-channel attacks)	V2X day one use cases (e.g., passive warnings and helpers)
3	+ Private ECU & private network	EAL 4+ (AVA_VAN.4 vulnerability resistance)	+ basic tamper resistance	Moderate	V2X box secure as stand alone device, but w/o trustworthy in-vehicle inputs	Safety relevant relying not only on V2X inputs
4	+ Relevant in-vehicle sensors and ECUs	EAL 4+ (AVA_VAN.5 vulnerability resistance)	+ moderate – high tamper resistance	Moderate – High	V2X box is trustworthy also regarding all relevant in-vehicle inputs	All

- Marko Wolf, – Hardware Security Modules for Protecting Automotive IT Systems – The EVITA project and beyond, escar USA 2013

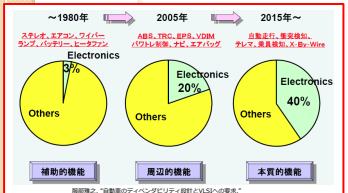
Vehicle Electrical System Security Committee

- SAEにて2011年から活動開始
  - 車載システムへの攻撃に関する研究発表などがトリガ
- 2つのタスクフォース
  - Automotive Security Guidelines and Risk Development
    - プロセスベースで車載システムのセキュリティレベルを策定
    - リスクを軽減するためのガイドラインおよび推奨デザインを作成
    - 2014年夏に初版リリースを目指す
  - Vehicle Electrical Hardware Security
    - ハードウェアベースのセキュリティ技術を利用
    - 「車載システムのセキュリティ」担保のための推奨デザインを作成
    - 2014年1月に作業完了予定

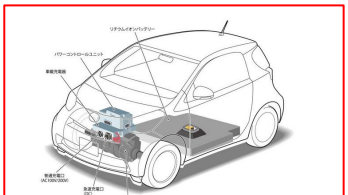
Battelle CyberAuto Challenge

- 自動車を中心とした交通システムに対する脅威と防御について学習
  - 高校/大学生向けサマーキャンプ: 1週間程度
  - USビッグ3やUS政府関係者などによる実地サポート
  - 実際に自動車をクラック
- 今年が第2回目

今後のクルマ



高度サービス・コスト削減のための電子化


電子部品比率が増えるEV

- 「計算機科学的アプローチ」によるセキュリティ技術が重要かつ必須
- 「EVならでは」も存在
  - EVは高トルク: エンジンチューンナップによる想定外の動き
  - 安価なサードパーティ製バッテリ: ただでさえEVは走行距離が短く、販売価格の30%を占めるバッテリ

まとめ

- クルマへの情報セキュリティ技術の必要性
  - 要因: 電子化比率の増加、標準化、EV化
- 気をつけること
  - ヒトの命を預かる耐久消費品
- 国内技術者の絶対的な不足

A.1.2　制御システム向け暗号の要件の考察

2013 年度 第 2 回軽量暗号 WG (2013 年 12 月 26 日) での日立製作所 大和田 徹氏による発表資料を示す。

HITACHI
Inspire the Next

CRYPTREC 軽量暗号WG

制御システム向け暗号の要件の考察

2013/12/26
(株)日立製作所 横浜研究所
大和田 徹

1　IT分野における情報セキュリティ

HITACHI
Inspire the Next

情報システムに対する様々なセキュリティ上の脅威が存在

■ 情報システムにおける主な脅威
- 情報漏えい、盗聴、なりすまし、不正アクセス
- データ/プログラムの改ざん、ウィルス感染
- DoS攻撃によるネットワーク/サーバのダウン、データ/プログラムの削除

■ 情報セキュリティ:
情報資産(保護資産)の定義 + 当該資産に対する3要素の維持

要素	定義	代表的な対策技術
機密性	認可されていない個人、エンティティまたはプロセスに対して、情報を使用不可または非公開にする特性	暗号、認証、アクセス制御
完全性	資産の正確さ、および完全さを保護する特性	改ざん検知、ログ管理、バックアップ
可用性	許可されたエンティティが要求したときに、アクセスおよび使用が可能である特性	ファイアウォール、リソースの多重化

情報システムでは各種対策技術を組合せて情報資産のセキュリティを確保

2 制御システム分野における情報セキュリティ優先事項

HITACHI Inspire the Next

制御システムにおける重要な保護資産とは

■ 制御システムと情報システムの比較(*)

比較項目	制御システム	情報システム
データ処理制約	リアルタイムかつ周期的な制御処理 ⇒遅延により制御不全に陥る可能性	処理集中による遅延は、 ある程度許容
システム更新頻度	10-20年	3-5年
稼動時間	24時間365日連続	一般には通常業務時間内
想定被害	システム不具合による人命影響の 可能性	金銭的損失、 プライバシー被害
優先保護資産	システムの連続稼動(可用性)	情報資産の漏洩防止(機密性)

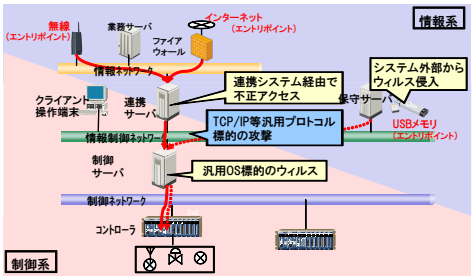
セキュリティ上の脅威に晒されても
制御システムが連続稼動すること(可用性)が最重要

＊IPA「重要インフラの制御システムセキュリティとITサービス継続に関する調査」を元に作成
http://www.ipa.go.jp/security/fy20/reports/ics-sec/

3 制御システムの一般的構成と脅威例

HITACHI Inspire the Next

昨今の制御システム～IT + CTのハイブリッド構成



物理アクセス性はシステム
によりまちまち
発電所、プラント～ビル
～スマートメータ

情報系部分は既存手法によるサイバー攻撃の対象となり得る
制御系部分への侵入で制御不全に

4 攻撃の顕在化と、それに対抗する動き

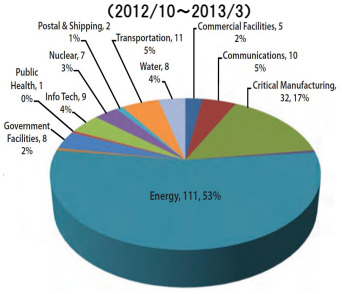
HITACHI Inspire the Next

制御システムへのサイバー攻撃が顕在化(12/下 200以上の報告),
制御システムに対するセキュリティ強化の要求が高まる

- WIB等, 業界レベルのセキュリティ規格策定段階から
  EDSA認証(CSSC)・CSMS認証(JIPDEC)等の
  国際的なセキュリティ認証制度整備段階へ進展
- ICS-CERTによる脆弱性公開が加速

コンポーネント/システムの
セキュリティ設計, セキュリティ運用
の重要性増大

重要インフラにおけるセキュリティ事故報告
(2012/10～2013/3)

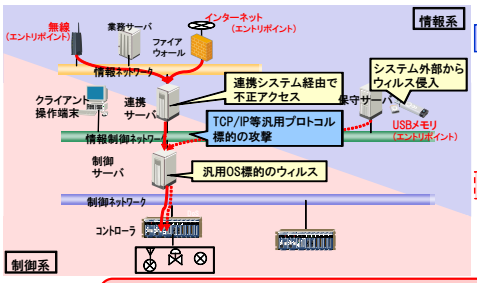


出典：米ICS-CERT,
http://ics-cert.us-cert.gov/sites/default/files/ICS-CERT_Monitor_April-June2013.pdf

5 制御システムにおけるセキュリティ対策

HITACHI Inspire the Next

**昨今の制御システム～IT + CTのハイブリッド構成
ITベース攻撃手法の対象となり得る**



対抗策
3要素を組合せた最新の
情報系セキュリティ対策技術

（大） （汎用・標準）
外部接続 採用技術
（小） （専用・独自）

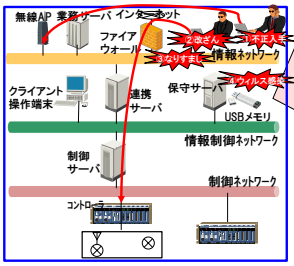
対抗策
制御系の特徴を考慮した
セキュリティ対策技術

制御系の特徴を考慮した対策技術の確立が急務

6 制御システムの可用性確保に向けた対策技術

HITACHI Inspire the Next

制御システムの可用性を損なう主要な脅威の洗い出し



可用性を損なう主要な脅威とその対策方針

#	脅威	対策方針
①	第三者がシステム内を流れるデータを不正に入手し、誤作動を引き起こすために悪用する(不正パケットの送信など)	パケットの内容が理解できないようにする
②	第三者がコントローラ宛のデータを改ざんし、コントローラが誤った制御指令をセンサに送信してしまう(誤動作、機能低下)	不正パケットを検知する
③	第三者が保守サーバになりすまして制御サーバを乗っ取り、コンローラの誤動作を引き起こす	通信相手が正しいことを確認する
④	保守員が保守サーバにアクセスする際、保守サーバに接続する可搬型記憶媒体(USBなど)がウィルスに感染していた為、ウィルスがシステム内に侵入し、プログラムやデータの改ざん/破壊、DoS攻撃を行う(システムを停止するため)	ウィルスを検知する

暗号技術が可用性確保に繋がる分野が存在

7 暗号適用時の課題

HITACHI Inspire the Next

制御系への暗号適用影響を分析し、課題を抽出

#	制御系の制約	暗号適用時に想定される影響	暗号適用時の課題	
1	周期的/リアルタイム処理	負荷増加により、周期的/リアルタイム処理の困難化	レイテンシ制約	性能
2	組込み機器のリソース制約(CPU性能、メモリ容量等)	高処理負荷暗号の実行によるシステム通常動作の困難化	暗号による処理負荷大	性能
3	全てがクリティカルな制御情報とは限らない	制御データ全てを暗号化対象とすると処理負荷大	不必要な暗号処理による不要な処理負荷の発生	性能/安全性
4	連続稼動	鍵管理/鍵更新処理がシステムの連続稼動に影響	連続稼動に影響しない鍵管理/鍵更新方式が確立されていない	運用
5	長期運用(10-20年)	暗号危殆化の可能性	暗号危殆化による安全性低下	運用

上記課題を解決する暗号機能の実現が求められている

＊ JPCERT/CC「重要社会インフラのためのプロセス制御システムのセキュリティ強化ガイド」を元に作成
www.jpcert.or.jp/research/2009/PCSSecGuide_20091120.pdf

8 制御システム向け暗号の要件

HITACHI Inspire the Next

課題から制御システム向け暗号の要件を抽出

#	【再掲】課題/課題の区分		制御システム向け暗号の要件
1	レイテンシ制約	性能	高速処理可能
2	暗号による処理負荷大	性能	低処理負荷
3	不必要な暗号処理による不要な処理負荷の発生	性能/安全性	柔軟な暗号化対象長
4	連続稼動に影響しない鍵管理/鍵更新方式が確立されていない	運用	低リソースな鍵管理/鍵更新機能
5	暗号危殆化による安全性低下	運用	複数鍵長/アルゴリズムの切替容易

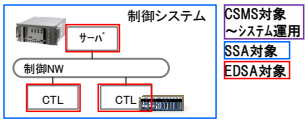
制御システム向けに上記要件を満たす「軽量暗号」が役立つ可能性はある

9 制御コンポーネントのセキュリティ認証

HITACHI Inspire the Next

動向

- IEC62443に則ったｾｷｭﾘﾃｨ評価認証ｽｷｰﾑ整備中
  - 事業・運用者向けCSMS認証(→IEC62443-2)
  - システムベンダ向けCSS認証(→IEC62443-3)
  - **装置ベンダ向けEDSA認証(→IEC62443-4)**
    - 将来的な調達要件化が想定
    - 総構造認証～システム認証は認証機器が前提



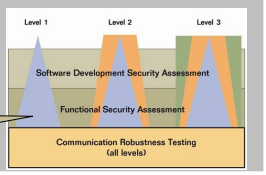
- CSSCがEDSA国際相互認証ｽｷｰﾑを準備中
  EDSA認証取得機器：2社4製品(‘13/12時点)

EDSA認証

ISAセキュリティ適合性協会(ISCI)が運営する制御機器のセキュリティ保証に関する認証制度

評価項目としては
1. 機能セキュリティアセスメント(FSA)
2. ソフトウェア開発セキュリティアセスメント(SDSA)
3. 通信ロバストネステスト(CRT)

の3項目が存在。



レベル1でのFSA要求は主にユーザ認証アクセスコントロール、データ完全性、機密性要求はレベル2以上で
→暗号化以前の課題が山積

出典：米ISA Secure
http://www.isasecure.org/ISASecure-Program.aspx

10 まとめ～制御システム向け暗号への期待

HITACHI Inspire the Next

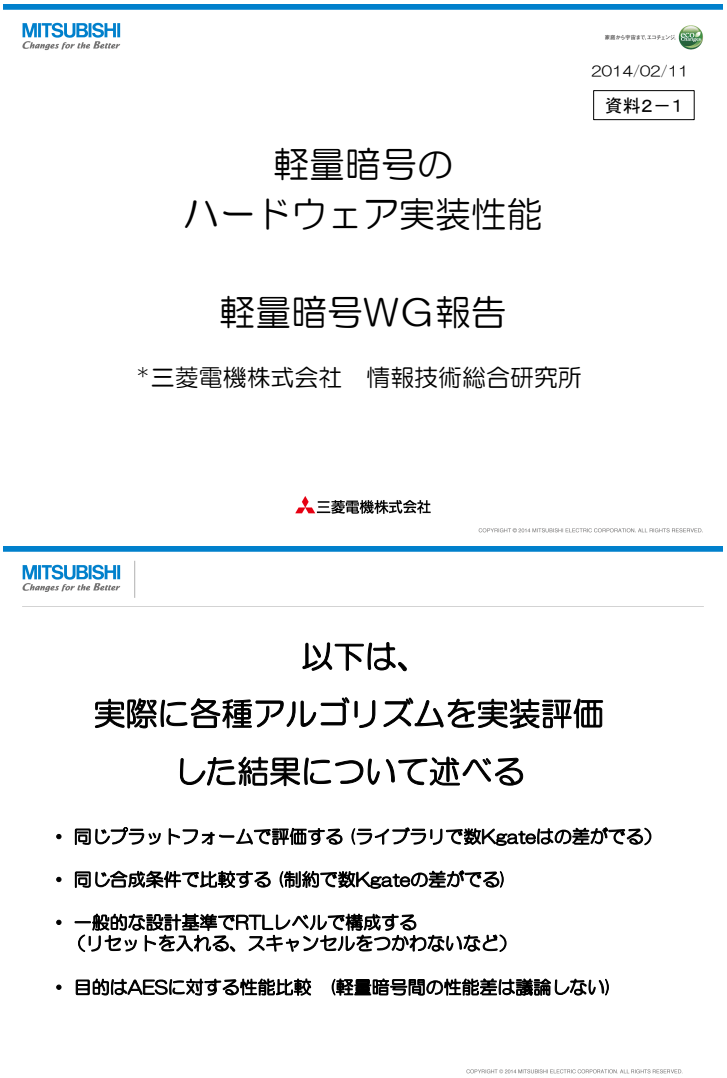
**実現したいシステムセキュリティからすると
暗号は数多くの要件のうちの一つ
～「グローバルで使い易い」暗号を期待**

- 暗号アルゴリズムだけ、では使いこなせない
  - 鍵管理手法等も含めたシステムパッケージを期待
  - 制御向けの暗号使い方ガイドを期待
- 制御コンポーネントは汎用技術活用の方向
  - 専用HWを必要としない組込みCPUに適した暗号を期待
- 評価認証スキームの確立と国際調達要件化
  - 国際標準でない暗号の採用困難化の方向→使える暗号の国際標準化推進を期待

A.2　軽量ブロック暗号の実装詳細評価

A.2.1　ハードウェア性能評価

2013 年度 第 3 回軽量暗号 WG (2014 年 2 月 20 日) での三菱電機 鈴木 大輔氏による発表資料を示す。



MITSUBISHI
Changes for the Better

2014/02/11

資料2－1

軽量暗号の
ハードウェア実装性能

軽量暗号WG報告

＊三菱電機株式会社　情報技術総合研究所

三菱電機株式会社

MITSUBISHI
Changes for the Better

以下は、
実際に各種アルゴリズムを実装評価
した結果について述べる

- 同じプラットフォームで評価する (ライブラリで数Kgateはの差がでる)
- 同じ合成条件で比較する (制約で数Kgateの差がでる)
- 一般的な設計基準でRTLレベルで構成する
  (リセットを入れる、スキャンセルをつかわないなど)
- 目的はAESに対する性能比較　(軽量暗号間の性能差は議論しない)

MITSUBISHI Changes for the Better

機能概略

F1. 鍵長は規定される最小のモードを想定する.

F2. 暗号化のみの実装とする.
(一部暗号化・復号も実装したので報告する)

F3. CPU のコプロセッサとしての利用を想定し，コンパクトで低電力とされるAPB バス接続が可能な設計とする.

MITSUBISHI Changes for the Better

設計方針概略

P1. 各アルゴリズムに対して3 種類の実装を行う:

(i) 典型的なround ベースの実装
(ii) 1 サイクルで処理が完了するunrolled 実装
(iii) データパスをS-box のサイズとするserial 実装

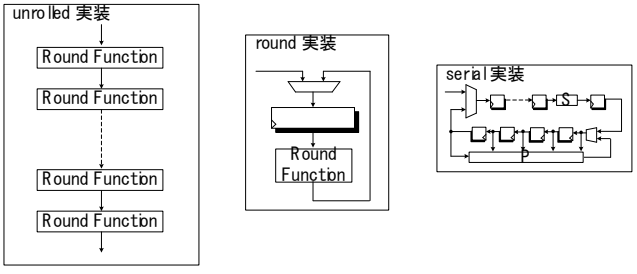
P2. 鍵スケジュールはon-the-yで実装する.

P3. CMOS セルライブラリを直接インスタンスするような最適化は行わず，ライブラリ非依存で合成可能な記述とする.

MITSUBISHI Changes for the Better

アーキテクチャ

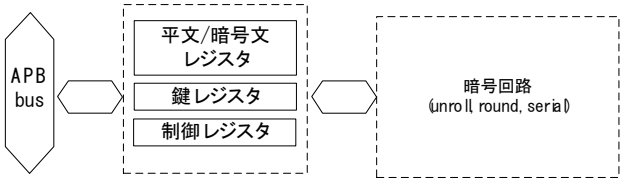
■実装方式



MITSUBISHI Changes for the Better

インターフェース

■インターフェース



MITSUBISHI Changes for the Better

評価環境

オープンセルライブラリと標準的なツールを使用.

論理合成ツール	Design Compiler (G-2012.06-SP5)
ライブラリ	NANGATE Open Cell Library(45nm CMOS)
合成制約	面積最小
遅延条件	NangateOpenCellLibrary slow (最悪条件の仮想遅延)
論理シミュレータ	NC-Verilog 10.20-s040
使用言語	Verilog-HDL

MITSUBISHI Changes for the Better

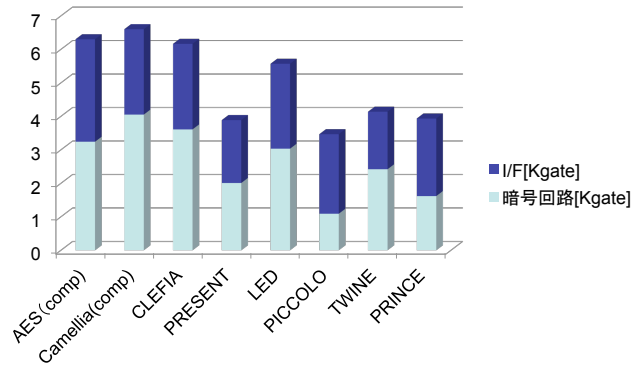
暗号化のみ

MITSUBISHI Changes for the Better

Serial

ゲートサイズ

AESに対して他のアルゴリズムは-2Kgateから+1Kgateの範囲

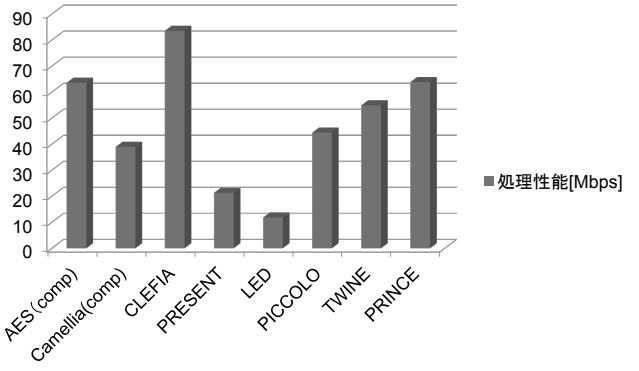


MITSUBISHI Changes for the Better

Serial

スループット

軽量暗号はSerial実装で高速化しにくい

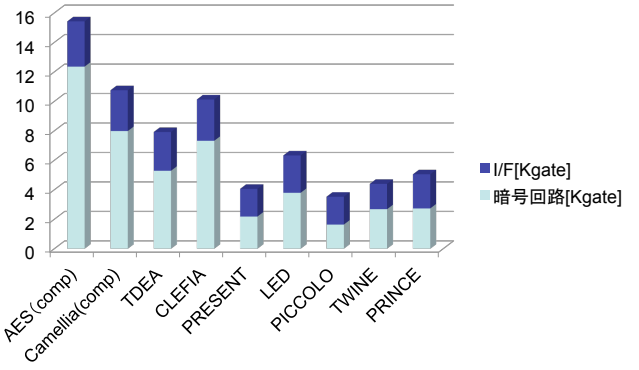


MITSUBISHI Changes for the Better

Round

ゲートサイズ

AESに対して軽量暗号はRound実装とSerial実装の回路規模に差がほとんどなく、4Kgate以下で実装できる

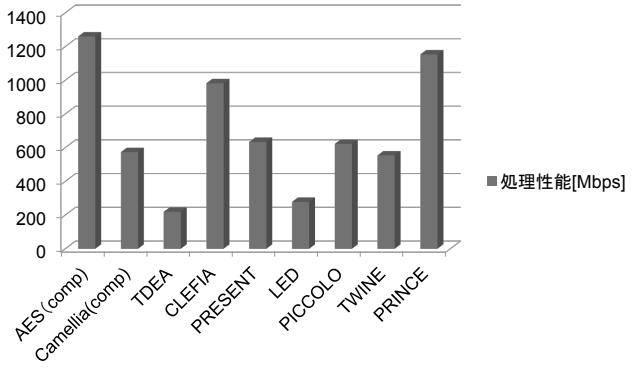


MITSUBISHI Changes for the Better

Round

スループット

軽量暗号はRound実装の軽量暗号はSerial実装のAESに対して速度性能約10倍

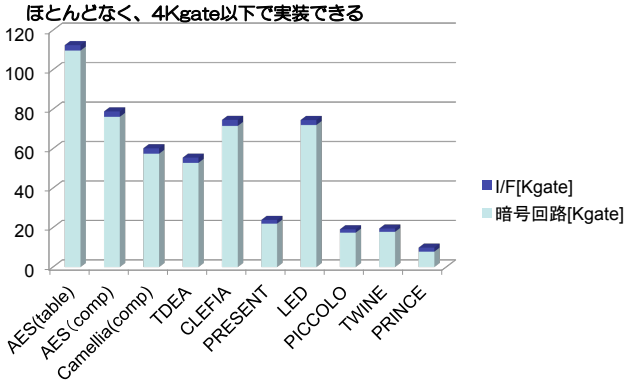


MITSUBISHI Changes for the Better

Unrolled

ゲートサイズ

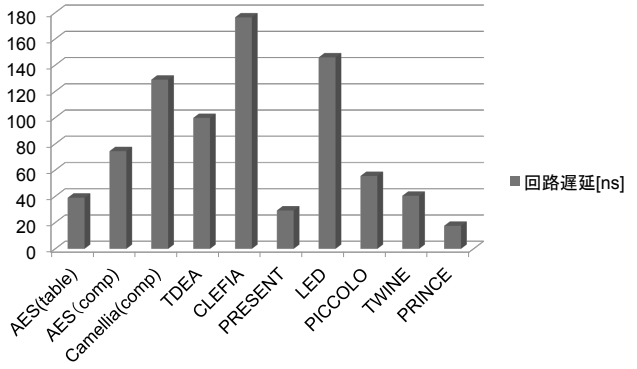
AESに対して軽量暗号はRound実装とSerial実装の回路規模に差がほとんどなく、4Kgate以下で実装できる



MITSUBISHI Changes for the Better

Unrolled

回路遅延



MITSUBISHI Changes for the Better

差分のまとめ

- 「論理回路性能の視点から」　軽量暗号とAESの違い

  ① 回路規模は１～2Kgate軽量暗号の方が小さい

  ② 約3Kgate以内でつくるなら軽量暗号の方がサイクル数が1/10

  ③ ②と同じサイクル数を達成するためにはAESは約10Kgate必要

  ④ 「1サイクル暗号化」に必要な回路規模が2ケタ違う。

  ⑤ 1サイクルとしてとれる周波数が2～3倍低遅延暗号が高速

MITSUBISHI Changes for the Better

RFIDをメインアプリと想定した場合

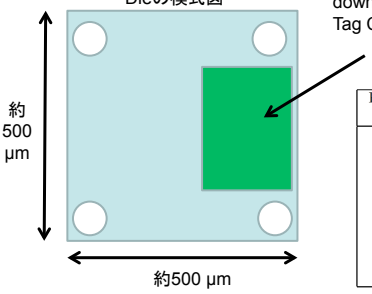
軽量暗号のハードウエア実装のアプリケーションと言えば「RFID」

この領域のアプリケーションでは処理時間は通信時間の方が支配的であることが多い

（②、③よりも）①の視点が産業上の有用性を見出せるか？

MITSUBISHI Changes for the Better

RFIDにおける実装制約



http://www.impinj.com/support/downloadable_documents.aspx#Monza 5 Tag Chips

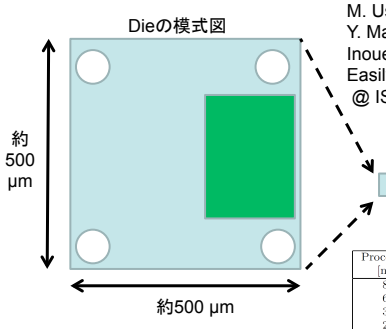
ロジック部は全体の3割程度

Process [nm]	Gate dencity [gate/mm^2]	500um 角の 30% 領域利用時 [gate]
800	1.2K	90
600	3K	225
350	18K	1.4K
250	35K	2.6K
180	80K	6K
130	180K	13.5K
90	400K	30K
65	800K	60K
40	1600K	120K

１～2Kgateの違いによって、実装の可否が変わるのは0.18 umくらいまで

MITSUBISHI Changes for the Better

RFIDにおける実装制約



M. Usami, H. Tanabe, A. Sato, I. Sakama, Y. Maki, T. Iwamatsu,T.Ipposhi and Y. Inoue:A 0.05 × 0.05 mm² RFID Chip with Easily Scaled-Down ID-Memory. @ ISSCC 2007

μ-chipくらいのRFIDであれば･･･

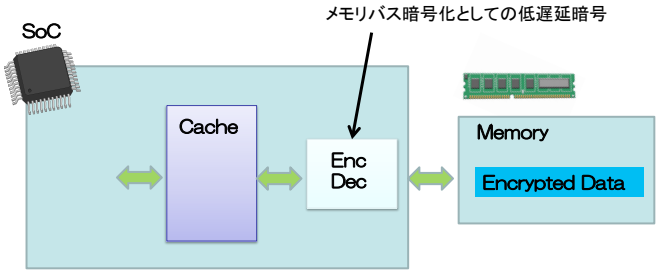
Process [nm]	Gate dencity [gate/mm²]	500um 角の 30% 領域利用時 [gate]	50um 角領域 利用時 [gate]
800	1.2K	90	3
600	3K	225	7.5
350	18K	1.4K	45
250	35K	2.6K	76
180	80K	6K	200
130	180K	13.5K	450
90	400K	30K	1K
65	800K	60K	2K
40	1600K	120K	4K

40nmでも1～2Kgateの差によって、実装の可否が変わる可能性あり

MITSUBISHI Changes for the Better

XOM, AEGIS

- 耐タンパプロセッサ
- 主記憶をOSや他のプロセス、あるいはプロービング攻撃などから秘匿することを目的として暗号化
- 読み出し速度がクリティカル



MITSUBISHI Changes for the Better

リアルタイム性能

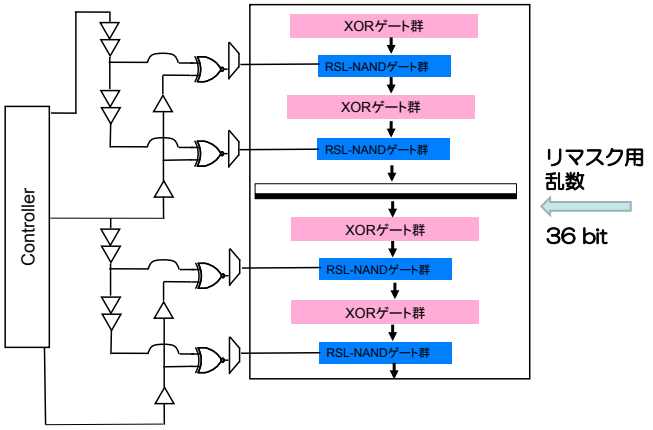
App.	Time region
Man-machine interface	数秒
Motor control	数 ms～数十 ms
I/O device control	数 μs
フラッシュ, EEPROM 読み出し	数 10～100ns
低電力 SRAM	数 10ns
高速 SRAM	～10ns

MITSUBISHI
Changes for the Better

＋　サイドチャネル対策

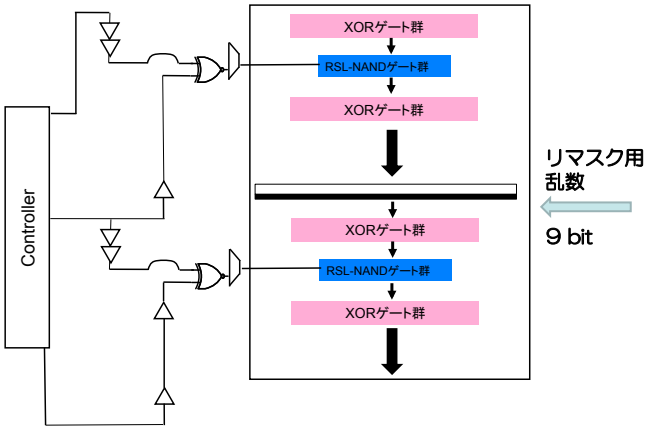
MITSUBISHI
Changes for the Better

8bit S-boxでのRSL



MITSUBISHI
Changes for the Better

4bit S-boxでのRSL



MITSUBISHI Changes for the Better

考察

■「軽量暗号」たる特徴は
- ✓ブロック長が64bit
- ✓鍵スケジュールが軽い（ラウンド定数のみ、レジスタ不要）
- ✓4bit S-box

これらがAESより1～2Kgate小さくなる主要因
(逆にいえば、これでほぼ特徴付けられる)

■ 改良は

暗号化のみ(PRESENT)
→復号もほぼ同じサイズでできる(Piccolo,TWINE)
→そもそも速い(低遅延 PRINCE)
→(認証暗号（FIDES）) という流れ？

MITSUBISHI Changes for the Better

まとめ

■ハードウェア実装からの視点としては軽量暗号は、
- ・マチュアなプロセスでの回路規模
- ・(リアルタイム)メモリ暗号化
- ・μ秒クラスのリアルタイム通信

などのアプリにおいて、AES に対してアドバンテージがある。

■小さい、速いという一つの指標だけだとAESとの差分が少ない。小さく、速く（低遅延)、サイドチャネル対策しやすい、のが良い軽量暗号、という考え方は？

■ファームウェア実装を考慮すれば、また違った視点
軽量暗号のアドバンテージが考えられる。

MITSUBISHI Changes for the Better

以下付録

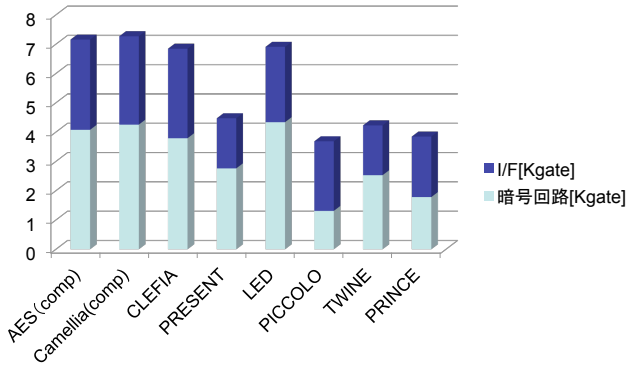
MITSUBISHI
Changes for the Better

暗号化・復号

MITSUBISHI
Changes for the Better

Serial

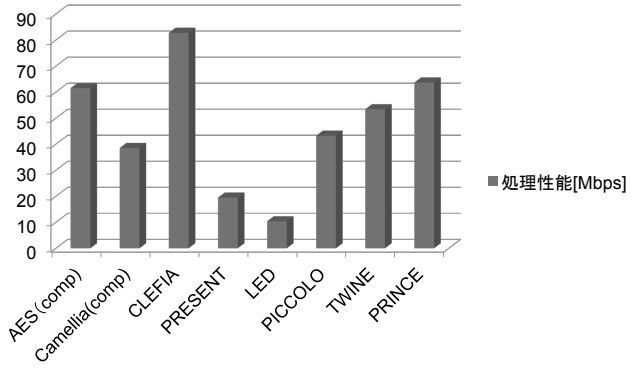
ゲートサイズ



MITSUBISHI
Changes for the Better

Serial

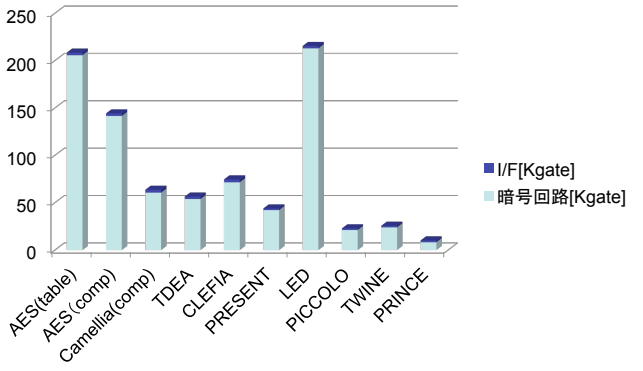
スループット



MITSUBISHI Changes for the Better

Unrolled

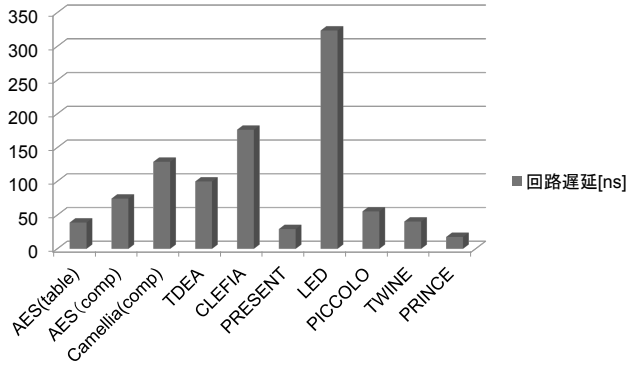
ゲートサイズ



MITSUBISHI Changes for the Better

Unrolled

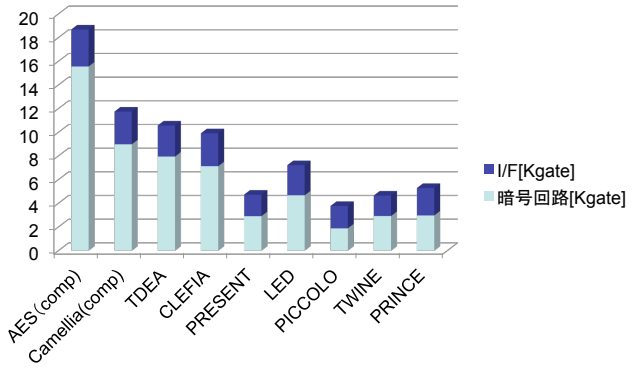
回路遅延



MITSUBISHI Changes for the Better

Round

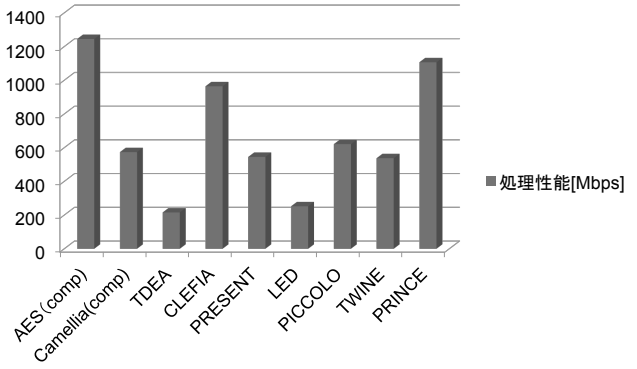
ゲートサイズ



MITSUBISHI Changes for the Better

Round

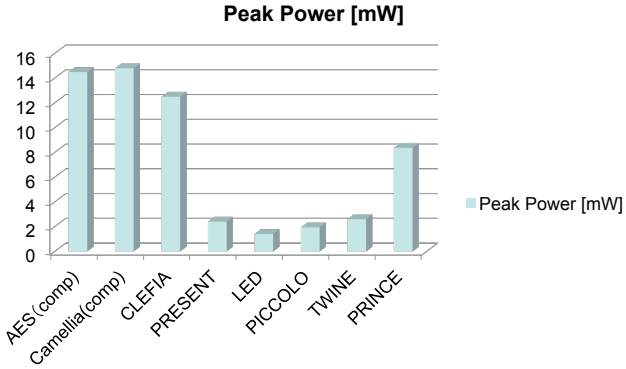
スループット



MITSUBISHI Changes for the Better

Serial

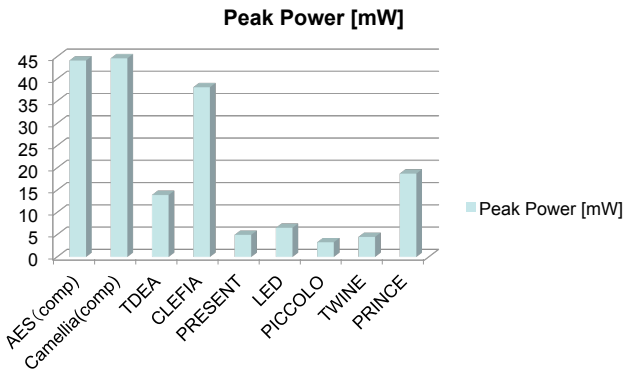
ピーク電流



MITSUBISHI Changes for the Better

Round

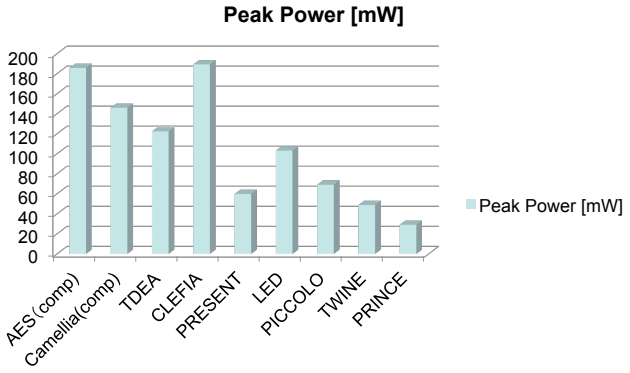
ピーク電流



MITSUBISHI Changes for the Better

Unrolled

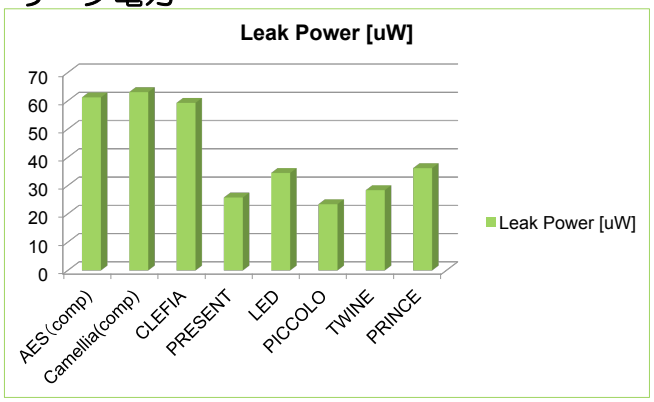
ピーク電流



MITSUBISHI Changes for the Better

Serial

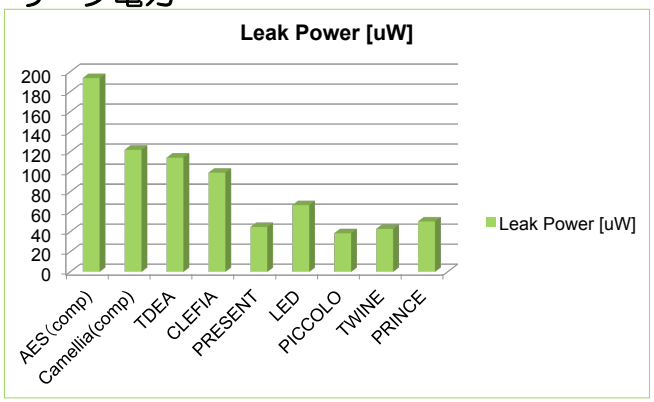
リーク電力



MITSUBISHI Changes for the Better

Round

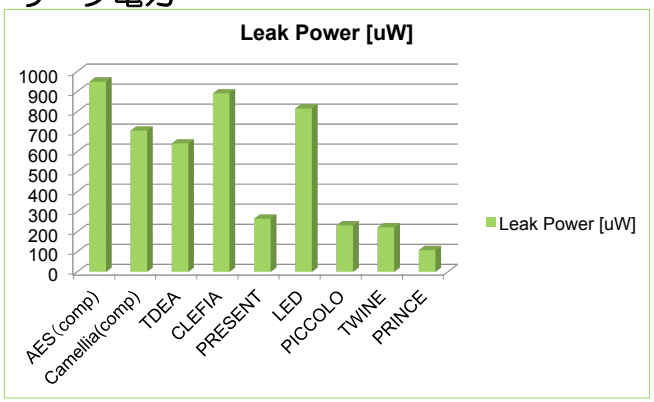
リーク電力



MITSUBISHI Changes for the Better

Unrolled

リーク電力



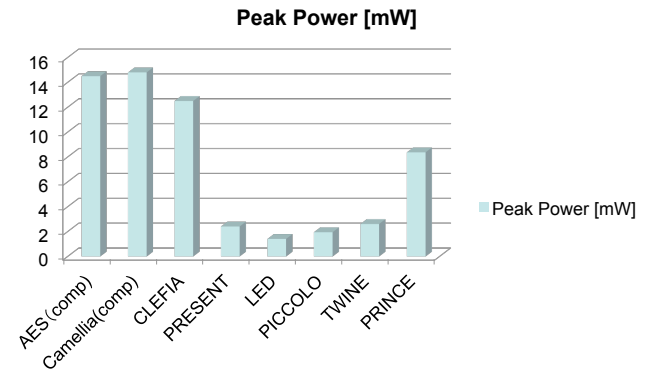
MITSUBISHI Changes for the Better

暗号化のみ(残りのデータ)

MITSUBISHI Changes for the Better

Serial

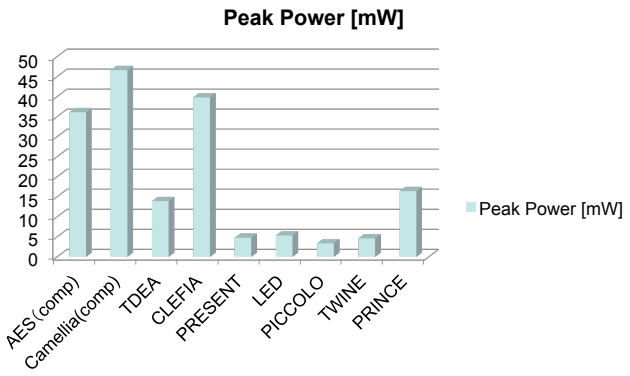
ピーク電流



MITSUBISHI Changes for the Better

Round

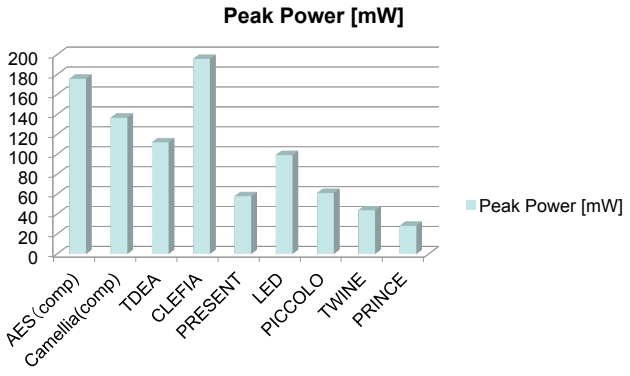
ピーク電流



MITSUBISHI Changes for the Better

Unrolled

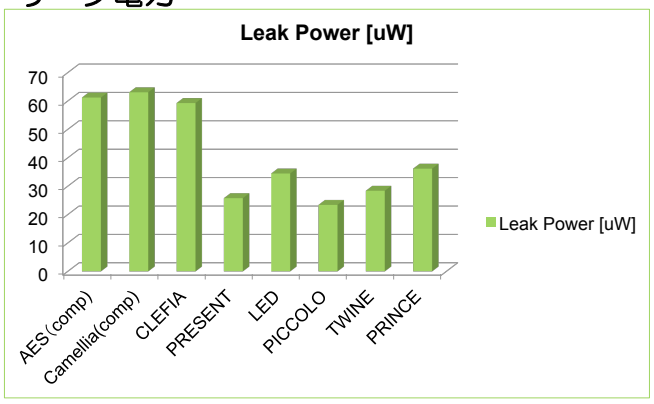
ピーク電流



MITSUBISHI Changes for the Better

Serial

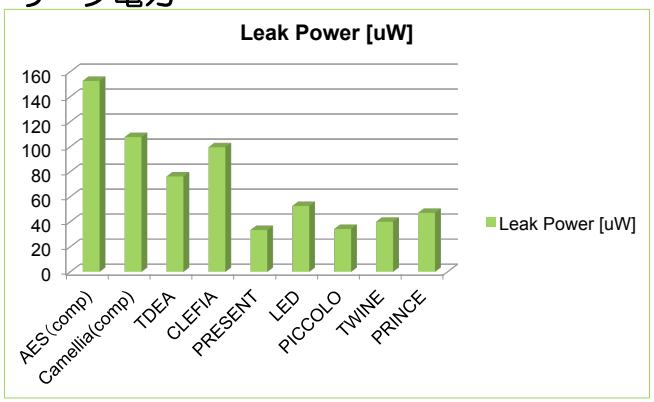
リーク電力



MITSUBISHI Changes for the Better

Round

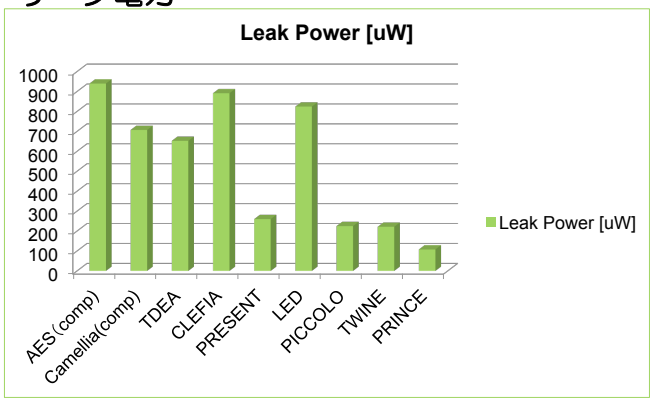
リーク電力



MITSUBISHI Changes for the Better

Unrolled

リーク電力



124

A.2.2　ソフトウェア性能評価

2013 年度 第 3 回軽量暗号 WG (2014 年 2 月 20 日) での三菱電機 松井 充氏による発表資料を示す。

資料2－2

軽量暗号のソフトウエア性能評価

(CRYPTREC軽量暗号WG資料)

2014年2月20日
三菱電機情報技術総合研究所　松井充

1

評価の目的

- Lightweight と呼ばれるブロック暗号がマイコン上のソフトウエアでどの程度 Lightweight であるかを調べる
  - Lightweight はハードウエアで語られることが多い
- マイコン上で小型を目指した暗号実装評価結果は数少ない
  - ソフトウエアでの暗号評価はほとんどの場合高速化が目標
  - しかも評価方法のコンセンサスがない
- 評価方法を提案
  - 実用的な観点からのインターフェースと評価方法を定義
  - ROM, RAM サイズを指定して、その範囲内で実装し速度を計測
  - 速度は無視してROMサイズがどこまで小さくなるかもみる

2

どの程度小さければLightweightか

- Renesas社マイコンRL78の場合
  - 汎用品（G1xシリーズ）： ROM 1KB, RAM 128B から
  - 車載品（F1xシリーズ）： ROM 8KB, RAM 512B から
- Atmel社マイコンAVRの場合
  - ATtiny: ROM 0.5KB, RAM 32B から ROM 16KB, RAM 1KB まで
  - ATtiny24/44/84 Automotive: ROM 2/4/8KB, RAM 128/256/512B
- 暗号機能はアプリケーションの一部
  - 暗号アルゴリズムが占有できるメモリ量は、通常全体のごく一部
  - 小さければ小さいほど暗号を使える品種が増える
- Lightweight というからには…
  - ROM 512B, RAM 64B 程度はめざしたいところ
  - この範囲の ROM, RAM サイズが議論されることは少ない

3

既存の評価事例 AES-128

独自インターフェース。C言語から呼び出し可能にするためは()内に示す追加メモリが必要

Algorithm	Processor	ROM	RAM static	RAM stack	Enc Speed	Dec Speed
AES (ED)	ATtiny	1659(+72)	33	0(+24)	4557n	7015n

http://perso.uclouvain.be/fstandae/lightweight_ciphers/ から作成

C言語から呼び出し可能。但し()内に示す平文・鍵領域やスタックがカウントされていない

Algorithm	Processor	ROM	RAM static	RAM stack	Enc Speed	Dec Speed
AES (ED)	ATmega	2070	176(+32)	0(+22)	2039+2555n	2039+6764n
AES (ED)	ATmega	2580	176(+32)	0(+22)	2039+2555n	2039+3193n

http://www.das-labor.org/wiki/AVR-Crypto-Lib/en から作成

C言語から呼び出し可能。RAMサイズには平文、鍵、スタックすべてを含む

Algorithm	Processor	ROM	RAM	Enc Speed	Dec Speed
AES (E)	RL78	486	78	7288n	-
AES (E)	RL78	1021	60	3855n	-
AES (ED)	RL78	970	84	7743n	1821+10862n
AES (ED)	RL78	1989	64	3917n	893+5911n

(E) Enc only
(ED) Enc+Dec
n: blocks
Size: bytes
Speed: cycles

Matsui, Murakami: FSE2013

4

既存の評価事例 Present-80

独自インターフェース。C言語から呼び出し可能にするためは()内に示す追加メモリが必要

Algorithm	Processor	ROM	RAM static	RAM stack	Enc Speed	Dec Speed
Present (ED)	ATtiny	1000(+72)	18	0(+24)	11342n	13599n

http://perso.uclouvain.be/fstandae/lightweight_ciphers/ から作成

上と同様の独自インターフェース。さらにカウントされていないスタックを加算

Algorithm	Processor	ROM	RAM static	RAM stack	Enc Speed	Dec Speed
Present (E)	ATtiny	204(+72)	18	0(+28)	190048n	-
Present (ED)	ATtiny	272(+72)	18	0(+30)	190048n	253384n
Present (E)	ATtiny	210(+72)	18	0(+28)	55784n	-
Present (ED)	ATtiny	278(+72)	18	0(+30)	55784n	77304n

http://rfidsec2013.iaik.tugraz.at/res/slides/Session4_Talk2_Verstegen.pdf から作成

C言語から呼び出し可能。RAMサイズには平文、鍵、スタックすべてを含む

Algorithm	Processor	ROM	RAM	Enc Speed	Dec Speed
Present (E)	RL78	210	54	144879n	-
Present (E)	RL78	897	42	9007n	-
Present (ED)	RL78	512	62	61634n	44068+60834n
Present (ED)	RL78	1855	48	9007n	1903+8920n

(E) Enc only
(ED) Enc+Dec
n: blocks
Size: bytes
Speed: cycles

Matsui, Murakami: FSE2013

5

評価方法

- インターフェースの統一が必要
  - 小型実装では、インターフェースの違いによるサイズ差は無視できない
  - 暗号を利用することによるすべてのオーバーヘッドを数値化すべき
- 実用性の観点から
  - 評価対象は高級言語から呼び出し可能なサブルーチンとして記述する
  - RAM サイズには平文や鍵の領域、スタックをすべて含める
  - アプリケーションプログラムの範囲をこえる特殊なことはしない
- 評価対象のソフトウエア仕様
  - 1ブロックを暗号化／復号する機能をもつ
  - 平文領域と暗号文領域は共通化する
  - 鍵領域は終了時に元の状態を復帰（一時的に変更してもよい）

6

評価対象と評価項目

- 評価対象

	AES	Camellia	Clefia	TDES	LED	Prince	Present	Piccolo	Twine
ブロックサイズ	128	128	128	64	64	64	64	64	64
鍵サイズ	128	128	128	168	128	128	80	80	80

- 評価環境
  - ルネサスマイコンRL78　CISCプロセッサで小型化に向いている
- 評価項目
  1. ROM 512B/1024B, RAM 64B/128B の4通り制約条件のもとで、暗号化のみの実装と、暗号化＋復号の実装をおこなう
  2. 暗号化のみで、ROM サイズを最小化する実装をおこなう
  3. ROM 2KB程度で、暗号化がどこまで高速になるかを調べる

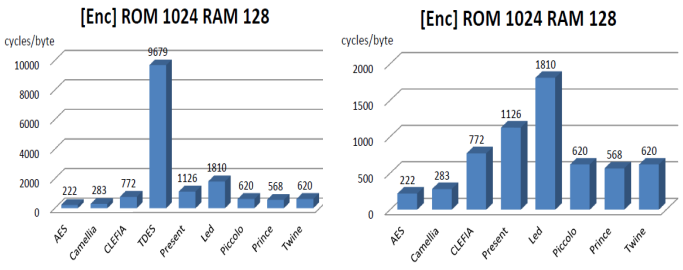
7

RL78 v.s. ATtiny

		RL78	ATtiny
ハードウエアレジスタ	レジスタ長	8, 16	8
	レジスタ数	8	32
アドレッシングモード	Read-Modify	Yes	No
	Post-Increment	No	Yes
命令長（bytes）	xor reg, [mem]	1-3	4
	call	3	2
	push / pop	1	2
実行時間（cycles）	read from RAM/ROM	1/4	2/3
	xor reg, [mem]	1	2
	taken/not-taken jump	4/2	2/1
	call + return	9	7

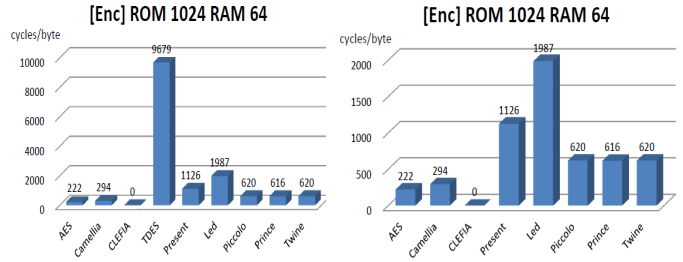
8

評価結果1: (E) ROM 1024B, RAM 128B



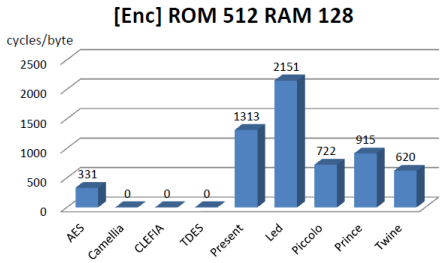
9

評価結果2: (E) ROM 1024B, RAM 64B



10

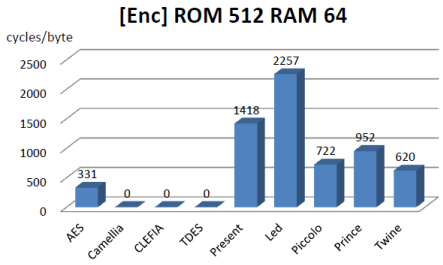
評価結果3: (E) ROM 512B, RAM 128B



11

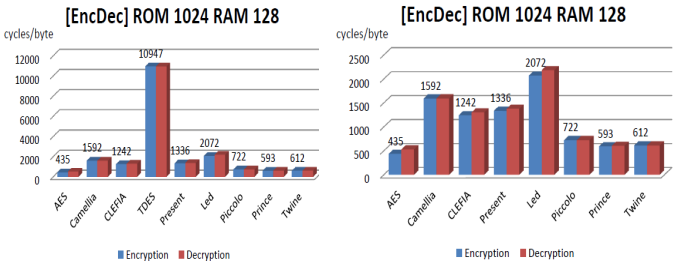
128

評価結果4：(E) ROM 512B, RAM 64B



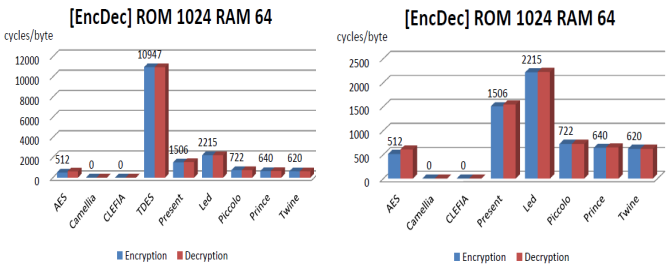
12

評価結果5：(ED) ROM 1024B, RAM 128B



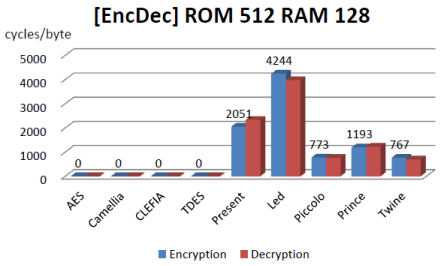
13

評価結果6：(ED) ROM 1024B, RAM 64B



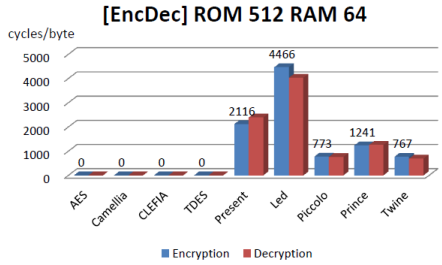
14

評価結果7: (ED) ROM 512B, RAM 128B

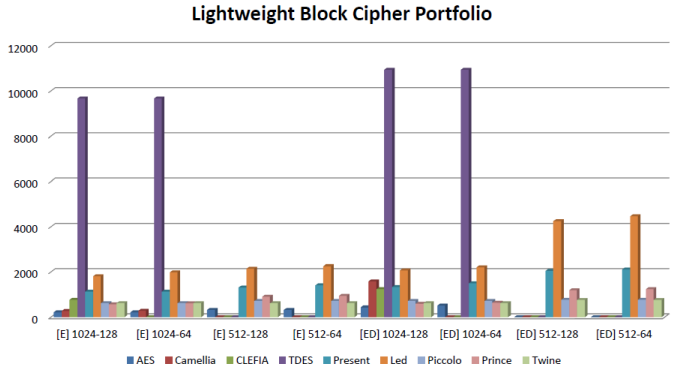


15

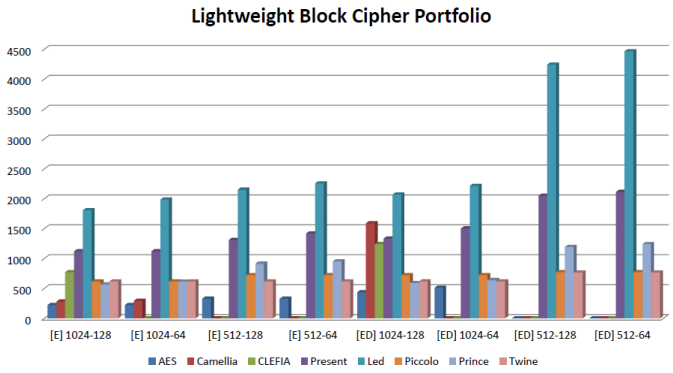
評価結果8: (ED) ROM 512B, RAM 64B



16

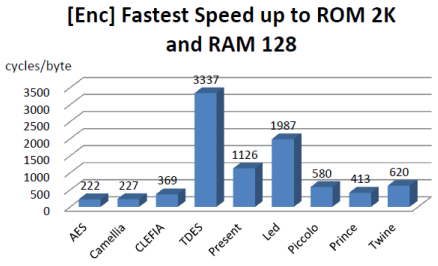


17



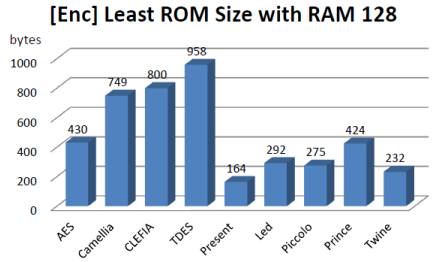
18

評価結果9：(E) Fastest Speed

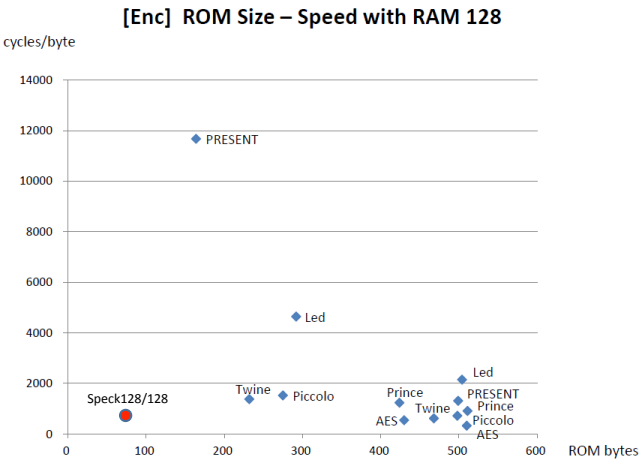


19

評価結果10：(E) Smallest Size



20



21

Lightweight v.s. AES

- アルゴリズム単体で考えるなら、暗号化のみならROM 512B、暗号復号両方こみならROM 1KBあれば、AESで十分
- 実際にはこれに加えモードを含む入出力データ処理が必要、またプロセッサのメモリ全てを暗号が使えるわけではない
- ROM 4KB-8KB, RAM 256B-512Bが、AESをソフトウエアで使えるプロセッサの下限と思われる。
- AESより価値あるソフトウエアLightweightブロック暗号とは…
  - メモリがたくさんあればAESなみの速度がでる
  - 暗号・復号こみでROM 200B以下、RAM 32B以下でそれなりの速度
  - 現時点ではNSAのSimon, Speckが有力候補（安全性は不明）

22

Software Lightweight Design

- 小型化実装は高速化実装と感覚がずいぶん違う
  - 無駄なコードを付け加えることが最終的に小型化に貢献することがある
  - 10バイト減らすと10倍遅くなることがある
- ほんの少しのことがコードサイズに大きく影響する
  - データの単なる移動や定数もオーバーヘッド
  - 数少ない単純な繰り返し構造だけでアルゴリズムを作る必要がある
- 鍵スケジュールがsoftware lightweightでない方式が多い
  - On-the-fly key schedulingを前提に設計すべき
- 回転シフト命令の効率はプロセッサに大きく依存
  - シフト命令もできるだけ避けよ
- Endian Neutralなアルゴリズムが望ましい
  - 今ではほとんどのプロセッサがlittle endianメモリアクセスなのに、多くのアルゴリズムがbig endianを前提に設計されている

23

その他私見

- 今回評価対象としたのはすべてS-box型ブロック暗号
- Lightweight ブロック暗号のトレンドは4ビットS-box
  - Present, LED, Prince, Piccolo, Twine
  - S-box型が安全性の評価がしやすい
- これは Lightweight として正しい方向か？
  - Simon, Speck が問うているもの
  - このタイプのブロック暗号TEAは昔からあった
- 暗号理論的にどこまで完全な安全性をめざすべきか？
  - 現実には side-channel attacks の方が脅威
  - そもそも64ビットブロック暗号がリバイバルしている
  - 安全性の条件を再定義する方向もあるのではないか

24